Symplectic schemes for highly oscillatory Hamiltonian systems: the homogenization approach beyond the constant frequency case

Matthew Dobson, Claude Le Bris, and Frédéric Legoll

August 6, 2010

Abstract

We follow up on our previous works [11, 12] which presented a possible approach for deriving symplectic schemes for a certain class of highly oscillatory Hamiltonian systems. The approach considers the Hamilton-Jacobi form of the equations of motion, formally homogenizes it and infers an appropriate symplectic integrator for the original system. In [11, 12] the case of a system exhibiting a single constant fast frequency was considered. The present work successfully extends the approach to systems that have either one *varying* fast frequency or *several constant* frequencies. Some related issues are also examined.

Contents

1 Introduction

The purpose of this work is to describe an approach for constructing symplectic numerical integrators for systems within the following class of highly oscillatory Hamiltonian systems:

$$\frac{dq}{dt} = \frac{\partial H_\varepsilon}{\partial p}(q,p), \quad \frac{dp}{dt} = -\frac{\partial H_\varepsilon}{\partial q}(q,p), \tag{1}$$

with

$$H_\varepsilon(q_1, q_2, p_1, p_2) = \frac{p_1^T p_1}{2} + \frac{p_2^T p_2}{2} + V(q_1, q_2) + \frac{q_2^T \Omega(q_1)^2 q_2}{2\varepsilon^2}, \tag{2}$$

where $\varepsilon \ll 1$ is a small parameter, where $q = (q_1, q_2) \in \mathbb{R}^s \times \mathbb{R}^f$ and $p = (p_1, p_2) \in \mathbb{R}^s \times \mathbb{R}^f$ (the upper indices s and f designate the slow and fast variables respectively), and where the interaction potential V and the fast frequency factor $\Omega(q_1)$ are all independent of ε. We assume that $\Omega(q_1)$ is a symmetric matrix whose eigenvalues are positive and bounded away from zero and that V is bounded from below. We also assume that the initial conditions for (1) scale with ε so that the initial energy is bounded independently of ε. Since the small parameter ε imparts fast frequencies on the model, direct numerical simulation of (1) to times of $O(1)$ and beyond is computationally expensive. This type of Hamiltonian system has already been studied in many works [2, 3, 8, 10, 14]. See [4, 12] for a short review of the literature, and [10, Chap. XIII and XIV] for a general overview. This paper continues the work initiated in [11, 12] by extending the generality of examples treated in the Hamilton-Jacobi framework. The major assumption remaining on the form of (2) is that the leading order behavior in the fast variables is that of a harmonic oscillator

$$\frac{p_2^T p_2}{2} + \frac{q_2^T \Omega(q_1)^2 q_2}{2\varepsilon^2}.$$

In Section 4, we briefly discuss the use of coordinate transforms for well-behaved, though fully non-linear, potentials [10, Sec. XIV.3]. The remainder of that section describes the use of our approach on a particular Hamiltonian whose fast potential does not take the form of a harmonic oscillator in the original coordinates.

As in our previous works [11, 12], our approach to the problem is based on the Hamilton-Jacobi formalism. Let $S_\varepsilon(t, q, P)$ be the solution to

$$\partial_t S_\varepsilon = H_\varepsilon\left(q + \partial_P S_\varepsilon, P\right), \quad S_\varepsilon(0, q, P) = 0. \tag{3}$$

For all (q, p, t), it is known that the functions $(Q(t), P(t))$, implicitly defined by

$$p = P(t) + \frac{\partial S_\varepsilon}{\partial q}(t, q, P(t)), \qquad Q(t) = q + \frac{\partial S_\varepsilon}{\partial P}(t, q, P(t)), \tag{4}$$

are solutions to (1) with initial conditions (q, p). For any approximation $\widetilde{S}_\varepsilon$ of the generating function and stepsize h, there is a corresponding symplectic map $\Psi_h : (q, p) \mapsto (Q(h), P(h))$ defined by the implicit relations

$$p = P(h) + \frac{\partial \widetilde{S}_\varepsilon}{\partial q}(h, q, P(h)), \qquad Q(h) = q + \frac{\partial \widetilde{S}_\varepsilon}{\partial P}(h, q, P(h)). \tag{5}$$

In the following we construct a function $\widetilde{S}_\varepsilon$ that approximates the solution of (3) for small t and ε. As long as we solve (5) exactly, this results in a symplectic numerical scheme (see [6]). Being able to generate a class of symplectic schemes motivates the strategy, which we use in the sequel, of making all *approximations* on the level of the generating function $\widetilde{S}_\varepsilon$ and from there solving (5) to build the numerical scheme. Of course, we cannot expect to solve these equations exactly, but we do so to high precision and observe good behavior with respect to preserving invariants.

We work within the parameter regime $\varepsilon \ll h \ll \varepsilon^{1/3}$. This yields a computational speed-up in comparison to standard algorithms such as velocity Verlet, where the time step must be taken smaller

than the characteristic time scale of the fast motion, $O(\varepsilon)$. However, we note that the energy preservation property of symplectic schemes is typically proven in the limit $h \to 0$ for a given Hamiltonian, and we are thus working outside the theoretical regime of symplectic schemes. We nonetheless choose symplecticity as a goal for our scheme and provide numerical tests of the preservation of energy and other invariants.

In contrast to [11, 12], where only the case of a system exhibiting *one single constant* fast frequency was considered, the present work successfully extends the approach to systems that have either *one varying* fast frequency, or *several constant* frequencies. Many of the results presented here have been announced in [5].

In addition to extending the approach to these two more general settings, we address one issue that we left pending in our previous works [11, 12]. This issue is related to the approximation of the high order derivatives present in the numerical scheme. One drawback of symplectic integrators constructed using the Hamilton-Jacobi approach is that they require high order derivatives of the Hamiltonian. The question then arises of whether or not these high order derivatives can be approximated by the corresponding finite differences without sacrificing the other advantages of the approach (symplecticity, accuracy, ...). We demonstrate here that it is indeed possible to use such finite difference approximations and keep all the features of the algorithm up to a chosen order.

The present article is organized as follows. In Section 2, we consider a Hamiltonian of the form

$$H_\varepsilon(q_1, q_2, p_1, p_2) = \frac{p_1^T p_1}{2} + \frac{p_2^T p_2}{2} + V(q_1, q_2) + \Omega(q_1)^2 \frac{q_2^T q_2}{2\varepsilon^2}, \tag{6}$$

where $\Omega(q_1)$ is a *scalar* fast frequency that *depends on the slow variables*. As was already mentioned for the constant frequency case in [12], the best option to implement our approach is to precondition the fast motion using a change of variables. This is to be performed prior to writing the Hamilton-Jacobi equation associated to the Hamiltonian dynamics. However, the dependency of Ω upon the slow variables introduces substantial new difficulties in this preconditioning step as compared to [12], which will be described in details below.

The algorithm that we obtain following this strategy has been introduced in [5]. In this paper we provide a comprehensive discussion of its derivation and properties as well as numerical tests demonstrating that the algorithm has favorable error performance in terms of resonances. The algorithm's computational efficiency is comparable, although slightly lower due to implicitness, to another established approach for small ε (such as 10^{-4}) and does not break down as $\varepsilon \to 0$. We further show that the exchange of actions (energy divided by frequency) among the fast degrees of freedom is captured well by our algorithm. The algorithm presented has many possible variants based on the choice of approximation of the generating function in (3). Among the possible variants, are those that reduce to the algorithms given in [12] whenever Ω is constant.

In Section 3, we next consider the case of a *non-scalar constant* fast frequency Ω,

$$H_\varepsilon(q_1, q_2, p_1, p_2) = \frac{p_1^T p_1}{2} + \frac{p_2^T p_2}{2} + V(q_1, q_2) + \frac{q_2^T \Omega^2 q_2}{2\varepsilon^2}. \tag{7}$$

We consider first the case when the fast frequencies present in Ω are non-resonant, and next address the case when some frequencies are resonant. These terms will be defined in Section 3. We will show the efficiency of the algorithm obtained on a classical test case.

We conclude our work by studying an example system composed of a single point attached by a spring to a pivot in the plane (see Section 4). Such a system cannot initially be written as a system with Hamiltonian of the form (2). After a change of coordinates, we transform the Hamiltonian into the form

$$H_\varepsilon(q_1, q_2, p_1, p_2) = \frac{p_1^T M(q_2)^{-1} p_1}{2} + \frac{p_2^T p_2}{2} + V(q_1, q_2) + \Omega^2 \frac{q_2^T q_2}{2\varepsilon^2}, \tag{8}$$

with a non-constant mass matrix $M(q_2)$. Form (8) is now compatible with (2), up to the fact that the mass matrix $M(q_2)$ depends on the fast position q_2. We show that our general approach again applies, yielding an algorithm with very good numerical performance.

Notation Before proceeding, we briefly fix the following notation. For vectors $u, v \in \mathbb{R}^n$, we define the dot product $u \cdot v \in \mathbb{R}$ by

$$u \cdot v = u^T v = \sum_{i=1}^{n} u_i v_i,$$

where we also use as shorthand $u^2 = u \cdot u$. For $u \in \mathbb{R}^m$ and $v \in \mathbb{R}^n$, we define the tensor product $u \otimes v \in \mathbb{R}^{m \times n}$ by

$$(u \otimes v)_{ij} = u_i v_j, \qquad \text{for } 1 \le i \le m, \quad 1 \le j \le n.$$

For matrices $A, B \in \mathbb{R}^{m \times n}$, we define $A : B \in \mathbb{R}$ by

$$A : B = \sum_{i=1}^{m} \sum_{j=1}^{n} A_{ij} B_{ij}.$$

We employ two different derivative notations. For functions $S(q_1, P_1)$ and $V(q_1, q_2)$ with $q_1, P_1 \in \mathbb{R}^s$ and $q_2 \in \mathbb{R}^f$, we denote the entries of q_1 by $q_1 = (q_{1,1}, \dots, q_{1,s})$, and we write $\partial_{q_1} S$ and $\nabla_1 V$ to denote the vectors

$$(\partial_{q_1} S)_j = \frac{\partial S}{\partial q_{1,j}}, \qquad (\nabla_1 V(q_1, q_2))_j = \frac{\partial V}{\partial q_{1,j}}(q_1, q_2) \qquad 1 \le j \le s.$$

2   The case of a scalar non-constant frequency

In this section, we consider the case (6) of a Hamiltonian with a scalar non-constant fast frequency $\Omega(q_1)$. Slightly changing the notation in comparison to (6), the Hamiltonian reads

$$H_\varepsilon(\check{q}_1, \check{q}_2, \check{p}_1, \check{p}_2) = \frac{\check{p}_1^T \check{p}_1}{2} + \frac{\check{p}_2^T \check{p}_2}{2} + \check{V}(\check{q}_1, \check{q}_2) + \Omega(\check{q}_1)^2 \frac{\check{q}_2^T \check{q}_2}{2\varepsilon^2}. \tag{9}$$

We first precondition the equations with a change of variables that takes into account the most oscillatory component of the fast solution. Following [10, page 555], we introduce the symplectic change of variables

$$q_1 = \check{q}_1, \qquad p_1 = \check{p}_1 - \frac{\nabla_1 \Omega(\check{q}_1)}{2\Omega(\check{q}_1)} \check{q}_2^T \check{p}_2, \qquad q_2 = \frac{\sqrt{\Omega(\check{q}_1)}}{\sqrt{\varepsilon}} \check{q}_2, \qquad p_2 = \frac{\sqrt{\varepsilon}}{\sqrt{\Omega(\check{q}_1)}} \check{p}_2, \tag{10}$$

which transforms (9) into

$$H_1(q_1, q_2, p_1, p_2) = \frac{1}{2}\left(p_1 + \frac{\nabla_1 \Omega(q_1) q_2^T p_2}{2\Omega(q_1)} \right)^2 + \frac{\Omega(q_1)}{2\varepsilon}(p_2^T p_2 + q_2^T q_2) + \check{V}\left(q_1, \frac{\sqrt{\varepsilon}}{\sqrt{\Omega(q_1)}} q_2 \right).$$

We neglect part of the slow momentum term and rewrite the slow potential to arrive at

$$H_2(q_1, q_2, p_1, p_2) = \frac{1}{2} p_1^T p_1 + \frac{\Omega(q_1)}{2\varepsilon}(p_2^T p_2 + q_2^T q_2) + V(q_1, \sqrt{\varepsilon} q_2),$$

where

$$V(q_1, q_2) = \check{V}\left(q_1, \frac{1}{\sqrt{\Omega(q_1)}} q_2 \right).$$

The system of equations corresponding to H_2 is

$$\dot{q}_1 = p_1, \qquad \dot{p}_1 = -\nabla_1 V(q_1, \sqrt{\varepsilon} q_2) - \frac{\nabla_1 \Omega(q_1)}{2\varepsilon}(p_2^T p_2 + q_2^T q_2)$$

$$\dot{q}_2 = \frac{\Omega(q_1)}{\varepsilon} p_2, \qquad \dot{p}_2 = -\sqrt{\varepsilon}\, \nabla_2 V(q_1, \sqrt{\varepsilon} q_2) - \frac{\Omega(q_1)}{\varepsilon} q_2$$

4

In the spirit of [12], we introduce the fast time

$$\sigma(t) = \frac{1}{\varepsilon} \int_0^t \Omega(q_1(s))\, ds \tag{11}$$

and consider the time-dependent change of variables $(q_2, p_2) \mapsto (\widetilde{x}, \widetilde{y})$ defined by

$$q_2 = \widetilde{x}\cos(\sigma(t)) + \widetilde{y}\sin(\sigma(t)), \qquad p_2 = -\widetilde{x}\sin(\sigma(t)) + \widetilde{y}\cos(\sigma(t)),$$

which captures the highest frequency component of the fast variables. The transformation is motivated by the fact that if V were independent of q_2, then $\widetilde{x}$ and $\widetilde{y}$ would be constant. This gives the dynamics

$$\dot{q}_1 = \frac{\partial H_3}{\partial p_1}, \qquad \dot{p}_1 = -\frac{\partial H_3}{\partial q_1} - \frac{\nabla_1 \Omega(q_1)}{2\varepsilon}(\widetilde{x}^T \widetilde{x} + \widetilde{y}^T \widetilde{y}),$$
$$\dot{\widetilde{x}} = \frac{\partial H_3}{\partial \widetilde{y}}, \qquad \dot{\widetilde{y}} = -\frac{\partial H_3}{\partial \widetilde{x}}, \tag{12}$$

where

$$H_3(q_1, \widetilde{x}, p_1, \widetilde{y}, t) = \frac{1}{2}p_1^2 + V\left[q_1, \sqrt{\varepsilon}(\widetilde{x}\cos\sigma(t) + \widetilde{y}\sin\sigma(t))\right].$$

Note that the dynamics on $(q_1, \widetilde{x}, p_1, \widetilde{y})$ includes a *memory* term due to the dependence of σ on the history of q_1, see (11). On the other hand, in the case of a constant frequency, Ω, there is no memory term since σ does not depend on q_1, and (12) is the Hamiltonian dynamics associated to H_3. In the present case, rather than operate on a system with memory, we consider σ as an additional variable and add the corresponding conjugate variable $\widetilde{a} = \frac{1}{2}(\widetilde{x}^T \widetilde{x} + \widetilde{y}^T \widetilde{y})$. We then have the dynamics

$$\dot{q}_1 = p_1,$$
$$\dot{p}_1 = -\nabla_1 V(q_1, \sqrt{\varepsilon}(\widetilde{x}\cos\sigma + \widetilde{y}\sin\sigma)) - \frac{\nabla_1 \Omega(q_1)}{\varepsilon}\widetilde{a},$$
$$\dot{\widetilde{x}} = \sqrt{\varepsilon}\sin\sigma\, \nabla_2 V(q_1, \sqrt{\varepsilon}(\widetilde{x}\cos\sigma + \widetilde{y}\sin\sigma)),$$
$$\dot{\widetilde{y}} = -\sqrt{\varepsilon}\cos\sigma\, \nabla_2 V(q_1, \sqrt{\varepsilon}(\widetilde{x}\cos\sigma + \widetilde{y}\sin\sigma)),$$
$$\dot{\sigma} = \frac{1}{\varepsilon}\Omega(q_1),$$
$$\dot{\widetilde{a}} = \sqrt{\varepsilon}\,(\widetilde{x}\sin\sigma - \widetilde{y}\cos\sigma)^T \nabla_2 V(q_1, \sqrt{\varepsilon}(\widetilde{x}\cos\sigma + \widetilde{y}\sin\sigma)),$$

which is a Hamiltonian dynamics associated with the energy

$$H_4(q_1, \widetilde{x}, \sigma, p_1, \widetilde{y}, \widetilde{a}) = \frac{1}{2}p_1^T p_1 + V(q_1, \sqrt{\varepsilon}(\widetilde{x}\cos\sigma + \widetilde{y}\sin\sigma)) + \widetilde{a}\frac{\Omega(q_1)}{\varepsilon}. \tag{13}$$

In the sequel, we take the above Hamiltonian H_4 as a starting point for our manipulations. We hence write the Hamilton-Jacobi equation associated to H_4 and then rescale the variables $\widetilde{a}, \widetilde{x}$, and $\widetilde{y}$,

$$a = \frac{\widetilde{a}}{\varepsilon}, \quad x = \frac{\widetilde{x}}{\sqrt{\varepsilon}}, \quad \text{and } y = \frac{\widetilde{y}}{\sqrt{\varepsilon}},$$

so that (q_1, x, P, Y, a) are of the same order. We choose not to rescale Σ despite the fact that it is $O(t/\varepsilon)$ at time t because it plays the role of a fast time in the original dynamics. The Hamilton-Jacobi equation becomes

$$\partial_t S_\varepsilon(t, q_1, x, \Sigma, P_1, Y, a) = H_4\left[q_1 + \partial_{P_1}S_\varepsilon, \sqrt{\varepsilon}\left(x + \frac{1}{\varepsilon}\partial_Y S_\varepsilon\right), \Sigma, P_1, \sqrt{\varepsilon}Y, \varepsilon a - \partial_\Sigma S_\varepsilon\right]$$
$$= \frac{1}{2}P_1^2 + V\left(q_1 + \partial_{P_1}S_\varepsilon, (\varepsilon x + \partial_Y S_\varepsilon)\cos\Sigma + \varepsilon Y\sin\Sigma\right)$$
$$+ \left(a - \frac{1}{\varepsilon}\partial_\Sigma S_\varepsilon\right)\Omega(q_1 + \partial_{P_1}S_\varepsilon), \tag{14}$$

$$S_\varepsilon(0, q_1, x, \Sigma, P_1, Y, a) = 0.$$

The equations corresponding to (4) are

$$Q_1 = q_1 + \partial_{P_1} S_\varepsilon, \qquad X = x + \frac{1}{\varepsilon} \partial_Y S_\varepsilon, \qquad \Sigma = \sigma + \frac{1}{\varepsilon} \partial_a S_\varepsilon,$$

$$P_1 = p_1 - \partial_{q_1} S_\varepsilon, \qquad Y = y - \frac{1}{\varepsilon} \partial_x S_\varepsilon, \qquad A = a - \frac{1}{\varepsilon} \partial_\Sigma S_\varepsilon, \tag{15}$$

where all derivatives of S_ε are evaluated at $(t, q_1, x, \Sigma, P_1, Y, a)$.

2.1 Expanding in ε

We wish to expand the generating function as a series in powers of ε. Recall that there are two balancing factors in choosing an ansatz. Since the expanded Hamilton-Jacobi equation of order ε^k will involve the generating function up to order ε^{k+1}, we need to make sufficiently strong assumptions to be able to separate out the $O(\varepsilon^{k+1})$ terms from the lower order terms into independent equations and thereby close the hierarchy. On the other hand, we also need to have a sufficiently general form in order to satisfy the resulting equations. We make the two-scale ansatz

$$S_\varepsilon(t, q_1, x, \Sigma, P_1, Y, a) = S_0(t, q_1, P_1, a) + \sum_{k=1}^\infty \varepsilon^k S_k \left(t, \Sigma - \frac{1}{\varepsilon} \tau(t, q_1, P_1, a), q_1, x, \Sigma, P_1, Y, a \right) \tag{16}$$

where we assume $S_k(t, \beta, q_1, x, \gamma, P_1, Y, a)$ is 2π periodic in γ and choose the fast time τ to satisfy

$$\partial_t \tau = \left(\nabla_1 V(q_1 + \partial_{P_1} S_0, 0) + a \nabla_1 \Omega(q_1 + \partial_{P_1} S_0) \right) \cdot \partial_{P_1} \tau + \Omega(q_1 + \partial_{P_1} S_0),$$

$$\tau(0, q_1, P_1, a) = 0. \tag{17}$$

The periodicity assumption on γ is needed to separate the orders in the resulting hierarchy of equations and is justified both by the fact that equation (14) is 2π-periodic in Σ and that the generating function of a harmonic oscillator is periodic in time. The simplifying choice of S_0 independent of Σ is consistent with (15) and the fact that A is $O(1)$ over long times. Similar arguments justify the choice that S_0 is a function of (t, q_1, P_1, a) only. Note that, as is common in homogenization expansions, we have introduced a fast time τ/ε in S_k, which is considered to be independent from the slow time t. Because of the non-constant frequency $\Omega(q_1)$, we choose for τ to depend on the slow variables (t, q_1, P_1, a). What is unusual is that we introduced the fast time as part of a transport term $\Sigma - \tau/\varepsilon$. We motivate this by briefly considering the Hamiltonian $H(a, \sigma) = -\varepsilon \cos(\sigma) - a/\varepsilon$, a highly simplified version of H_4 (see (13)). It is periodic in σ and keeps the conjugate variable a, with frequency $\Omega = 1$. In this case, the generating function $S_\varepsilon(t, a, \Sigma) = -at/\varepsilon - \varepsilon^2[\sin \Sigma - \sin(\Sigma - t/\varepsilon)]$ satisfies the Hamilton-Jacobi equation $\partial_t S = H(a + \partial_\Sigma S, \Sigma)$ with initial condition $S(0, a, \Sigma) = 0$. Inspired by the transport structure in $\sin(\Sigma - t/\varepsilon)$, we choose the independent variables Σ and $\Sigma - \tau/\varepsilon$ for S_k, rather than Σ and τ/ε.

In view of (16), we then have

$$\begin{array}{llll}
\partial_t S_\varepsilon &=& \sum_{k=0}^\infty \varepsilon^k (\partial_t S_k - \partial_t \tau \, \partial_\beta S_{k+1}), & \partial_Y S_\varepsilon &=& \sum_{k=1}^\infty \varepsilon^k \partial_Y S_k, \\
\partial_{P_1} S_\varepsilon &=& \sum_{k=0}^\infty \varepsilon^k (\partial_{P_1} S_k - \partial_{P_1} \tau \, \partial_\beta S_{k+1}), & \partial_\Sigma S_\varepsilon &=& \sum_{k=1}^\infty \varepsilon^k (\partial_\beta S_k + \partial_\gamma S_k).
\end{array}$$

From the initial condition $S_\varepsilon(0, q_1, x, \Sigma, P_1, Y, a) = 0$ and the ansatz, we have

$$S_0(0, q_1, P_1, a) = 0 \quad \text{and} \quad S_k(0, \gamma, q_1, x, \gamma, P_1, Y, a) = 0 \quad \text{for } 1 \le k < \infty, \tag{18}$$

where we note the repetition of arguments in S_k results from (16) and the fact that $\tau(0, q_1, P_1, a) = 0$.

2.1.1 Order ε^0 and ε

Inserting the ansatz (16) into (14) and expanding in terms of ε, the $O(\varepsilon^0)$ and $O(\varepsilon^1)$ equations are

$$O(1): \quad \partial_t S_0 - \partial_t \tau \, \partial_\beta S_1 = \frac{1}{2} P_1^T P_1 + \widehat{V} + (a - \partial_\beta S_1 - \partial_\gamma S_1) \widehat{\Omega},$$

$$O(\varepsilon): \quad \partial_t S_1 - \partial_t \tau \, \partial_\beta S_2 = \nabla_1 \widehat{V} \cdot (\partial_{P_1} S_1 - \partial_{P_1} \tau \partial_\beta S_2)$$

$$+ \nabla_2 \widehat{V} \cdot ((x + \partial_Y S_1) \cos\gamma + Y \sin\gamma)$$

$$+ (a - \partial_\beta S_1 - \partial_\gamma S_1) \nabla_1 \widehat{\Omega} \cdot (\partial_{P_1} S_1 - \partial_{P_1} \tau \, \partial_\beta S_2) - (\partial_\beta S_2 + \partial_\gamma S_2) \widehat{\Omega},$$

where

$$\widehat{V} = V(q_1 + \partial_{P_1} S_0 - \partial_{P_1} \tau \, \partial_\beta S_1, 0) \quad \text{and} \quad \widehat{\Omega} = \Omega(q_1 + \partial_{P_1} S_0 - \partial_{P_1} \tau \, \partial_\beta S_1).$$

As $\widehat{V}$ and $\widehat{\Omega}$ depend on S_1, we must proceed carefully in closing the equations. In fact, we will show $S_1 = 0$ and close the equations on S_0 by expanding the unknowns S_0 and S_1 in powers of t.

Lemma 2.1. *The $O(1)$ and $O(\varepsilon)$ equations imply*

$$\partial_t S_0 = \frac{1}{2} P_1^T P_1 + V(q_1 + \partial_{P_1} S_0, 0) + a\Omega(q_1 + \partial_{P_1} S_0), \tag{19}$$

$$S_1 = 0. \tag{20}$$

Proof. We expand the functions S_0, S_1 and τ in powers of t:

$$S_0 = S_{0,0} + t S_{0,1} + \frac{t^2}{2} S_{0,2} + \cdots, \quad S_1 = S_{1,0} + t S_{1,1} + \frac{t^2}{2} S_{1,2} + \cdots, \quad \tau = \tau_0 + t\tau_1 + \frac{t^2}{2}\tau_2 + \cdots$$

where from the initial conditions of (17) and (18), we have that $S_{0,0} = \tau_0 = 0$. The $O(\varepsilon^0 t^0)$ equation is

$$S_{0,1} - \tau_1 \partial_\beta S_{1,0} = \frac{1}{2} P_1^T P_1 + V(q_1, 0) + (a - \partial_\beta S_{1,0} - \partial_\gamma S_{1,0}) \Omega(q_1).$$

We wish to separate the S_0 and S_1 terms. We first observe that (17) gives $\tau_1 = \Omega(q_1)$ which leaves

$$S_{0,1} = \frac{1}{2} P_1^T P_1 + V(q_1, 0) + (a - \partial_\gamma S_{1,0}) \Omega(q_1).$$

We integrate both sides of the equation with respect to γ between 0 and 2π and note that, by the periodicity assumption in the ansatz, $\int_0^{2\pi} \partial_\gamma S_{1,0} \, d\gamma = 0$. Since S_0 is independent of γ, this leaves

$$S_{0,1} = \frac{1}{2} P_1^T P_1 + V(q_1, 0) + a\Omega(q_1),$$

which is (19) up to $O(t)$ error. This implies $\partial_\gamma S_{1,0} = 0$ which, combined with the initial condition on S_1, implies $S_{1,0} = 0$, so that (20) holds up to $O(t)$ error.

From here we proceed by induction. Suppose that (19) and (20) hold up to $O(t^k)$. We note from (17) that

$$\partial_{P_1} \tau = O(t^2),$$

and thus $\partial_{P_1} \tau \, \partial_\beta S_1 = O(t^{k+2})$. We then infer from the $O(1)$ equation that

$$\partial_t S_0 - \frac{t^k}{k!} \tau_1 \partial_\beta S_{1,k} = \frac{1}{2} P_1^T P_1 + V(q_1 + \partial_{P_1} S_0, 0) + a\Omega(q_1 + \partial_{P_1} S_0)$$

$$- \frac{t^k}{k!} (\partial_\gamma S_{1,k} + \partial_\beta S_{1,k}) \Omega(q_1) + O(t^{k+1}).$$

Upon canceling out the two $\partial_\beta S_{1,k}$ terms using $\tau_1 = \Omega(q_1)$ and integrating with respect to γ, we find that (19) holds up to $O(t^{k+1})$ and that $\partial_\gamma S_{1,k} = 0$.

Because S_1 vanishes up to an error of $O(t^k)$, we infer from (17) that

$$\partial_t \tau - \left(\nabla_1 \widehat{V} + a\nabla_1 \widehat{\Omega}\right) \cdot \partial_{P_1} \tau - \widehat{\Omega} = O(t^{k+2}).$$

We first use this in the $O(\varepsilon)$ equation to cancel the $\partial_\beta S_2$ terms. We then neglect all terms of $O(t^k)$ to find

$$\frac{t^{k-1}}{(k-1)!} S_{1,k} = \nabla_2 V(q_1 + \partial_{P_1} S_0, 0) \cdot (x\cos\gamma + Y\sin\gamma) - \Omega(q_1 + \partial_{P_1} S_0)\partial_\gamma S_2 + O(t^k).$$

Integrating this equation with respect to γ gives $S_{1,k} = 0$. Thus, we have shown that (20) holds up to $O(t^{k+1})$. Consequently, we have shown by induction that (19) and (20) hold to all orders. $\qquad\square$

In view of (20), we have the following simplified expressions for $\widehat{V}$ and $\widehat{\Omega}$:

$$\widehat{V} = V(q_1 + \partial_{P_1} S_0, 0) \quad \text{and} \quad \widehat{\Omega} = \Omega(q_1 + \partial_{P_1} S_0).$$

In the proof to Lemma 2.1 we have found $S_{0,1} = \frac{1}{2} P_1^T P_1 + V(q_1, 0) + a\Omega(q_1)$. Using (19), we likewise find $S_{0,2} = \nabla_1 V(q_1, 0) \cdot P_1 + a\nabla_1 \Omega(q_1) \cdot P_1$. For our generating function, we keep these two lowest order terms. We combine them, neglecting only $O(t^3)$ terms, to give

$$\widetilde{S}_0 = t\left[\frac{1}{2} P_1^T P_1 + V\left(q_1 + \frac{t}{2} P_1, 0\right) + a\Omega\left(q_1 + \frac{t}{2} P_1\right)\right]. \tag{21}$$

We likewise keep the first two orders in the fast time τ. From (17), we have $\tau_0 = 0$, $\tau_1 = \Omega(q_1)$, and $\tau_2 = \nabla\Omega(q_1) \cdot P_1$. We combine these, neglecting only $O(t^3)$ terms, and take the approximation

$$\widetilde{\tau} = t\Omega\left(q_1 + \frac{t}{2} P_1\right). \tag{22}$$

Inserting $S_1 = 0$ into the $O(\varepsilon)$ equation and using equation (17) leaves

$$\widehat{\Omega}\,\partial_\gamma S_2 = \nabla_2 \widehat{V} \cdot (x\cos\gamma + Y\sin\gamma). \tag{23}$$

2.1.2   Order ε^2

The $O(\varepsilon^2)$ equation of (14) is

$$
\begin{aligned}
\partial_t S_2 - \partial_t \tau\,\partial_\beta S_3 = {}&\frac{1}{2}(\nabla_{11}\widehat{V} + a\nabla_{11}\widehat{\Omega}) : (\partial_{P_1}\tau \otimes \partial_{P_1}\tau)(\partial_\beta S_2)^2 \\
&+ (\nabla_1 \widehat{V} + a\nabla_1 \widehat{\Omega}) \cdot (\partial_{P_1} S_2 - \partial_{P_1}\tau\,\partial_\beta S_3) \\
&- \nabla_{12}\widehat{V} : (\partial_\beta S_2\,\partial_{P_1}\tau \otimes \ell) + \frac{1}{2}\nabla_{22}\widehat{V} : \ell\otimes\ell + \nabla_2\widehat{V}\cdot\partial_Y S_2 \cos\gamma \\
&+ (\partial_\gamma S_2 + \partial_\beta S_2)\,\nabla_1\widehat{\Omega}\cdot(\partial_{P_1}\tau\,\partial_\beta S_2) - (\partial_\gamma S_3 + \partial_\beta S_3)\,\widehat{\Omega}
\end{aligned}
\tag{24}
$$

where $\ell = x\cos\gamma + Y\sin\gamma$. Since our approximation $\widetilde{\tau}$ given in (22) satisfies (17) up to $O(t^2)$, we can cancel the terms involving $\partial_\beta S_3$ up to $O(t^2)$, which leaves

$$
\begin{aligned}
\partial_t S_2 = {}&\frac{1}{2}(\nabla_{11}\widehat{V} + a\nabla_{11}\widehat{\Omega}) : (\partial_{P_1}\widetilde{\tau} \otimes \partial_{P_1}\widetilde{\tau})(\partial_\beta S_2)^2 + (\nabla_1 \widehat{V} + a\nabla_1 \widehat{\Omega}) \cdot \partial_{P_1} S_2 \\
&- \nabla_{12}\widehat{V} : (\partial_\beta S_2\,\partial_{P_1}\widetilde{\tau} \otimes \ell) + \frac{1}{2}\nabla_{22}\widehat{V} : \ell\otimes\ell + \nabla_2\widehat{V}\cdot\partial_Y S_2 \cos\gamma \\
&+ (\partial_\gamma S_2 + \partial_\beta S_2)\,\nabla_1\widehat{\Omega}\cdot(\partial_{P_1}\widetilde{\tau}\,\partial_\beta S_2) - \partial_\gamma S_3\,\widehat{\Omega} + O(t^2).
\end{aligned}
$$

As before, we integrate with respect to γ between 0 and 2π to remove the $\partial_\gamma S_3$ term, leaving only an equation in S_2. This time, however, S_2 is not itself independent of γ. From (23) we can write

$$S_2(t, \beta, q_1, x, \gamma, P_1, Y, a) = \frac{\nabla_2 \widehat{V}}{\widehat{\Omega}} \cdot (x\sin\gamma - Y\cos\gamma) + C(t, \beta, q_1, x, P_1, Y, a).$$

8

The initial condition $S_2(0, \gamma, q_1, x, \gamma, P_1, Y, a) = 0$ implies that

$$C(0, \beta, q_1, x, P_1, Y, a) = -\frac{\nabla_2 V(q_1, 0)}{\Omega(q_1)} \cdot (x \sin \beta - Y \cos \beta). \tag{25}$$

We integrate (24) with respect to γ between 0 and 2π, which gives

$$\partial_t C = \nabla_{22} \widehat{V} : \frac{(x \otimes x + Y \otimes Y)}{4} - \frac{(\nabla_2 \widehat{V})^2}{2\widehat{\Omega}} + (\nabla_1 \widehat{V} + a \nabla_1 \widehat{\Omega}) \cdot \partial_{P_1} C$$
$$+ \left[\nabla_1 \widehat{\Omega} \cdot \partial_{P_1} \widetilde{\tau} + \frac{1}{2}(\nabla_{11} \widehat{V} + a \nabla_{11} \widehat{\Omega}) : (\partial_{P_1} \widetilde{\tau} \otimes \partial_{P_1} \widetilde{\tau}) \right] (\partial_\beta C)^2 + O(t^2) \tag{26}$$

where we have used

$$\partial_t C = \oint_0^{2\pi} \partial_t S_2 \, d\gamma, \qquad \partial_\beta C = \oint_0^{2\pi} \partial_\beta S_2 \, d\gamma, \qquad \partial_{P_1} C = \oint_0^{2\pi} \partial_{P_1} S_2 \, d\gamma,$$

$$\oint_0^{2\pi} \partial_Y S_2 \cos \gamma \, d\gamma = \oint_0^{2\pi} \left[-\frac{\nabla_2 \widehat{V}}{\widehat{\Omega}} \cos^2 \gamma + \partial_Y C \cos \gamma \right] d\gamma = -\frac{\nabla_2 \widehat{V}}{2\widehat{\Omega}},$$

and

$$\oint_0^{2\pi} \frac{1}{2} \nabla_{22} \widehat{V} : \ell \otimes \ell \, d\gamma = \frac{1}{2} \nabla_{22} \widehat{V} : \oint_0^{2\pi} (x \otimes x \cos^2 \gamma + x \otimes Y \cos \gamma \sin \gamma + Y \otimes Y \sin^2 \gamma) \, d\gamma$$
$$= \frac{1}{4} \nabla_{22} \widehat{V} : (x \otimes x + Y \otimes Y).$$

We expand

$$C(t, \beta, q_1, x, P_1, Y, a) = C_0(\beta, q_1, x, P_1, Y, a) + t C_1(\beta, q_1, x, P_1, Y, a) + \frac{t^2}{2} C_2(\beta, q_1, x, P_1, Y, a) + \cdots$$

and insert this expansion into (26). The $O(t)$ term satisfies

$$C_1 = \frac{1}{4} \nabla_{22} V(q_1, 0) : (x \otimes x + Y \otimes Y) - \frac{(\nabla_2 V(q_1, 0))^2}{2\Omega(q_1)}.$$

We keep the $O(t^0)$ term (25) along with the $O(t)$ term to approximate S_2 up to $O(t^2)$ error with

$$\widetilde{S}_2 = \frac{\nabla_2 V(q_1 + t P_1, 0)}{\Omega(q_1 + t P_1)} \cdot (x \sin \gamma - Y \cos \gamma) - \frac{\nabla_2 V(q_1, 0)}{\Omega(q_1)} \cdot (x \sin \beta - Y \cos \beta)$$
$$+ t \left[\frac{1}{4} \nabla_{22} V(q_1, 0) : (x \otimes x + Y \otimes Y) - \frac{(\nabla_2 V(q_1, 0))^2}{2\Omega(q_1)} \right]. \tag{27}$$

2.2 Generating function and algorithm

Combining our previous approximations (21) and (27), we approximate the solution to (14) by

$$\widetilde{S}_\varepsilon = \widetilde{S}_0 + \varepsilon^2 \widetilde{S}_2$$
$$= t \left[\frac{1}{2} P_1^T P_1 + V \left(q_1 + \frac{t}{2} P_1, 0 \right) + a \Omega \left(q_1 + \frac{t}{2} P_1 \right) \right]$$
$$+ \varepsilon^2 \left[\frac{\nabla_2 V(q_1 + t P_1, 0)}{\Omega(q_1 + t P_1)} (x \sin \Sigma - Y \cos \Sigma) \right.$$
$$- \frac{\nabla_2 V(q_1, 0)}{\Omega(q_1)} \left(x \sin \left(\Sigma - \frac{t}{\varepsilon} \Omega(q_1 + \frac{t}{2} P_1) \right) - Y \cos \left(\Sigma - \frac{t}{\varepsilon} \Omega(q_1 + \frac{t}{2} P_1) \right) \right)$$
$$+ \frac{t}{4} \nabla_{22} V(q_1, 0) : (x \otimes x + Y \otimes Y) - t \frac{(\nabla_2 V)^2}{2\Omega} \right], \tag{28}$$

9

where we note
$$S_\varepsilon(t) = \widetilde{S}_\varepsilon + O(t^3) + O(\varepsilon^2 t^2) + O(\varepsilon^3).$$

The existence of derivatives of V in the generating function is problematic because it leads to higher-order gradient computations in the resulting numerical scheme. In many cases of interest, these higher-order gradients are computationally expensive. Furthermore, computing them potentially involves extra effort during implementation as most traditional schemes only involve first derivatives of V. To avoid this issue in our numerical scheme, we replace all derivatives of V found in the generating function with finite differences. We make this replacement before inserting the generating function into (5) in order to maintain the symplecticity of the scheme. Note that there is no unique way to form the finite differences.

In doing this, we choose to exclude $-\dfrac{(\nabla_2 V)^2}{2\Omega}$ from the $O(\varepsilon^2 t)$ term as it is much more computationally expensive to implement as a finite difference than the other terms. Rather, we only retain a portion of the $O(\varepsilon^2 t)$ term that turns out to be sufficient to reproduce the exchange of fast actions (see the definition (33) and simulation in Fig. 3 below). Note that the only change in the order of accuracy is caused by not retaining the portion of the $O(\varepsilon^2 t)$ term mentioned above. We choose the approximation

$$
\begin{aligned}
S_{\varepsilon,\mathrm{FD}} = {}& t\left[\frac{1}{2}P_1^T P_1 + V\left(q_1 + \frac{t}{2}P_1, 0\right) + a\Omega\left(q_1 + \frac{t}{2}P_1\right)\right] \\
&+ \frac{\varepsilon}{\Omega(q_1 + tP_1)}\left[V(q_1 + tP_1, \varepsilon(x\sin\Sigma - Y\cos\Sigma)) - V(q_1 + tP_1, 0)\right] \\
&+ \frac{\varepsilon}{\Omega(q_1)}\left[V(q_1, 0) - V\left(q_1, \varepsilon\left(x\sin(\Sigma - \frac{t}{\varepsilon}\Omega(q_1 + \frac{t}{2}P_1)) - Y\cos(\Sigma - \frac{t}{\varepsilon}\Omega(q_1 + \frac{t}{2}P_1))\right)\right)\right] \\
&+ \frac{t}{4}\left[V(q_1, \varepsilon x) - 2V(q_1, 0) + V(q_1, -\varepsilon x) + V(q_1, \varepsilon Y) - 2V(q_1, 0) + V(q_1, -\varepsilon Y)\right].
\end{aligned}
$$

Substituting $S_{\varepsilon,\mathrm{FD}}$ into (5), we arrive at the following expressions, where we use $\sigma = \Sigma - \dfrac{h}{\varepsilon}\Omega\left(q_1 + \dfrac{h}{2}P_1\right)$.

10

We first solve for (P_1, Y, Σ) in the implicit equations

$$
\begin{aligned}
P_1 = {}& p_1 - h[\nabla_1 V(q_1 + \frac{h}{2} P_1, 0) + a\nabla_1 \Omega(q_1 + \frac{h}{2} P_1)] \\
& - \varepsilon h \nabla_1 \Omega(q_1 + \frac{h}{2} P_1) \frac{\nabla_2 V(q_1, \varepsilon(x \sin \sigma - Y \cos \sigma))}{\Omega(q_1)} [x \cos \sigma + Y \sin \sigma] \\
& - \varepsilon \frac{\nabla_1 V(q_1 + hP_1, \varepsilon(x \sin \Sigma - Y \cos \Sigma)) - \nabla_1 V(q_1 + hP_1, 0)}{\Omega(q_1 + hP_1)} \\
& + \varepsilon \frac{\nabla_1 \Omega(q_1 + hP_1)(V(q_1 + hP_1, \varepsilon(x \sin \Sigma - Y \cos \Sigma)) - V(q_1 + hP_1, 0))}{\Omega(q_1 + hP_1)^2} \\
& - \varepsilon \frac{\nabla_1 V(q_1, 0) - \nabla_1 V(q_1, \varepsilon(x \sin \sigma - Y \cos \sigma))}{\Omega(q_1)} \\
& + \varepsilon \frac{\nabla_1 \Omega(q_1)(V(q_1, 0) - V(q_1, \varepsilon(x \sin \sigma - Y \cos \sigma)))}{\Omega(q_1)^2} \qquad\qquad (29) \\
& - \frac{h}{4}\Big(\nabla_1 V(q_1, \varepsilon x) - 2\nabla_1 V(q_1, 0) + \nabla_1 V(q_1, -\varepsilon x) \\
& \qquad\quad + \nabla_1 V(q_1, \varepsilon Y) - 2\nabla_1 V(q_1, 0) + \nabla_1 V(q_1, -\varepsilon Y)\Big), \\
Y = {}& y - \varepsilon \frac{\nabla_2 V(q_1 + hP_1, \varepsilon(x \sin \Sigma - Y \cos \Sigma))}{\Omega(q_1 + hP_1)} \sin \Sigma + \varepsilon \frac{\nabla_2 V(q_1, \varepsilon(x \sin \sigma - Y \cos \sigma))}{\Omega(q_1)} \sin \sigma \\
& - \frac{h}{4}\Big(\nabla_2 V(q_1, \varepsilon x) - \nabla_2 V(q_1, -\varepsilon x)\Big), \\
\Sigma = {}& \sigma + \frac{h}{\varepsilon} \Omega(q_1 + \frac{h}{2} P_1).
\end{aligned}
$$

We next compute (Q_1, X, A) using

$$
\begin{aligned}
Q_1 = {}& q_1 + hP_1 + \frac{h^2}{2}\Big[\nabla_1 V(q_1 + \frac{h}{2} P_1, 0) + a\nabla_1 \Omega(q_1 + \frac{h}{2} P_1)\Big] \\
& + \varepsilon \frac{h^2}{2} \nabla_1 \Omega(q_1 + \frac{h}{2} P_1) \frac{\nabla_2 V(q_1, \varepsilon(x \sin \sigma - Y \cos \sigma))}{\Omega(q_1)} [x \cos \sigma + Y \sin \sigma] \\
& + h\varepsilon \frac{\nabla_1 V(q_1 + hP_1, \varepsilon(x \sin \Sigma - Y \cos \Sigma)) - \nabla_1 V(q_1 + hP_1, 0)}{\Omega(q_1 + hP_1)} \\
& - h\varepsilon \frac{(V(q_1 + hP_1, \varepsilon(x \sin \Sigma - Y \cos \Sigma)) - V(q_1 + hP_1, 0))\nabla_1 \Omega(q_1 + hP_1)}{\Omega(q_1 + hP_1)^2}, \\
X = {}& x - \varepsilon \frac{\nabla_2 V(q_1 + hP_1, \varepsilon(x \sin \Sigma - Y \cos \Sigma))}{\Omega(q_1 + hP_1)} \cos \Sigma + \varepsilon \frac{\nabla_2 V(q_1, \varepsilon(x \sin \sigma - Y \cos \sigma))}{\Omega(q_1)} \cos \sigma \qquad (30) \\
& + \frac{h}{4}\Big(\nabla_2 V(q_1, \varepsilon Y) - \nabla_2 V(q_1, -\varepsilon Y)\Big), \\
A = {}& a + \varepsilon \frac{\nabla_2 V(q_1, \varepsilon(x \sin \sigma - Y \cos \sigma))}{\Omega(q_1)} [x \cos \sigma + Y \sin \sigma] \\
& - \varepsilon \frac{\nabla_2 V(q_1 + hP_1, \varepsilon(x \sin \Sigma - Y \cos \Sigma))}{\Omega(q_1 + hP_1)} [x \cos \Sigma + Y \sin \Sigma],
\end{aligned}
$$

where we recall $\sigma = \Sigma - \frac{h}{\varepsilon} \Omega\left(q_1 + \frac{h}{2} P_1\right)$. The equations (29) are implicit in $Z = (P_1, Y, \Sigma)$ and can be written in the form

$$
Z = z + hF(Z) + \varepsilon G(Z) + \frac{h}{\varepsilon} K(Z) = z + \mathcal{A}(Z), \qquad\qquad (31)
$$

11

with $z = (p_1, y, \sigma)$ and where $K(Z) = (0, 0, \Omega(q_1 + \frac{h}{2}P_1))$ (the dependency of F, G and K on (q_1, x, a) is not explicitly written). We solve (31) for Z with a simple fixed point iteration. In our simulations the gradient of $\mathcal{A}$ with respect to Z is small since the parameters h and ε are small and since in practice we work with $h^2/\varepsilon < 0.4$ (see e.g. Fig. 4 below). We thus expect the fixed point algorithm to quickly converge. In the test case considered below, this is indeed the case. We terminate the iteration when successive iterates differ by a relative factor of 10^{-10} or less, and the algorithm typically converges in 3 to 8 iterations, depending on the stepsize h and the parameter ε.

The equations (10), (29) and (30) yield Algorithm 1 outlined below.

Algorithm 1. Initialize: From the initial conditions $(\check{q}_1(0), \check{q}_2(0), \check{p}_1(0), \check{p}_2(0))$, compute the initial conditions

$$q_1(0) = \check{q}_1(0), \qquad\qquad p_1(0) = \check{p}_1(0) - \frac{\nabla_1 \Omega(\check{q}_1(0))}{2\Omega(\check{q}_1(0))} \check{q}_2(0)^T \check{p}_2(0),$$

$$x(0) = \frac{\sqrt{\Omega(\check{q}_1(0))}}{\varepsilon} \check{q}_2(0), \qquad y(0) = \frac{1}{\sqrt{\Omega(\check{q}_1(0))}} \check{p}_2(0),$$

$$\sigma(0) = 0, \qquad\qquad a(0) = \frac{1}{2\Omega(\check{q}_1)} \left(\check{p}_2^T(0)\check{p}_2(0) + \frac{\Omega(\check{q}_1)^2}{\varepsilon^2} \check{q}_2^T(0)\check{q}_2(0) \right).$$

Iterate: for $n \geq 0$,

1. Set $(q_1, x, \sigma, p_1, y, a) = (q_1^n, x^n, \sigma^n, p_1^n, y^n, a^n)$.

2. Solve the implicit equations (29) for (P_1, Y, Σ).

3. Compute (Q_1, X, A) using (30).

4. Set $\left(q_1^{n+1}, x^{n+1}, \sigma^{n+1}, p_1^{n+1}, y^{n+1}, a^{n+1}\right) = (Q_1, X, \Sigma, P_1, Y, A)$.

Post-process: From $\left(q_1^N, x^N, \sigma^N, p_1^N, y^N, a^N\right)$, return to the original variables: compute first (q_2^N, p_2^N) using

$$q_2^N = \sqrt{\varepsilon} \left[x^N \cos(\sigma^N) + y^N \sin(\sigma^N) \right], \quad p_2^N = \sqrt{\varepsilon} \left[-x^N \sin(\sigma^N) + y^N \cos(\sigma^N) \right],$$

and compute next $(\check{q}_1^N, \check{q}_2^N, \check{p}_1^N, \check{p}_2^N)$ using

$$\check{q}_1^N = q_1^N, \qquad \check{p}_1^N = p_1^N + \frac{\nabla_1 \Omega(q_1^N)}{2\Omega(q_1^N)} q_2^N \cdot p_2^N, \qquad \check{q}_2^N = \frac{\sqrt{\varepsilon}}{\sqrt{\Omega(q_1^N)}} q_2^N, \qquad \check{p}_2^N = \frac{\sqrt{\Omega(q_1^N)}}{\sqrt{\varepsilon}} p_2^N.$$

As a variant, we will also consider a "no-loop" version of Algorithm 1. Instead of solving (31) for Z, we perform the following update:

$$z^\star = z + hF(z) + \frac{h}{\varepsilon}K(z),$$

$$Z^\star = z + hF(z^\star) + \varepsilon G(z^\star) + \frac{h}{\varepsilon}K(z^\star), \tag{32}$$

and approximate the solution $Z = (P_1, Y, \Sigma)$ to (31) (namely, (29)) by $Z^\star = (P_1^\star, Y^\star, \Sigma^\star)$. Using $Z^\star$, we next determine (Q_1, X, A) in an explicit fashion, using (30). This yields an explicit scheme (we first determine $z^\star$, next $Z^\star$ and finally (Q_1, X, A)), with a lower computational cost than (29)-(30). However, this scheme is not symplectic since we do not fully solve (5). In Fig. 5 below, we will observe that the no-loop version has comparable error behavior with a reduced computational cost, though it seems difficult to justify why the energy and invariants are well preserved in this version.

12

2.3 Numerical results

In this section, we provide numerical tests of the behavior of our algorithm. As mentioned above, we are not interested in computing exact trajectories for the fast variables. Rather, we are interested in how well the algorithm preserves invariants of the system as well as its prediction of quantities derived from the fast variables. This system undergoes a slow exchange between the actions (33) of the fast degrees of freedom, and we test how well the algorithm captures this exchange. We also test the robustness of the algorithm by testing for the appearance of resonances.

We compare our algorithm to a well-known integrator for highly oscillatory dynamical systems, the Mollify algorithm [7], which is a modification of the previously proposed Impulse (also known as Verlet-I/r-RESPA) algorithm [9, 15]. Both Impulse and Mollify follow a kick/oscillate/kick pattern and incorporate the slow forces $\nabla \check{V}$ only at the "kick" steps, which are separated by a macro time step that is large with respect to the shortest timescale in the solution. This time step is typically larger than ε and is thus larger than the stable regime for Verlet. For the "oscillate" step, these methods integrate the fast forces using a stepsize that is small with respect to ε. The Mollify algorithm differs from the Impulse algorithm in how it incorporates the forces at the "kick" steps in order to improve the stability of the algorithm in the face of resonances, an issue we discuss later. These algorithms are designed to minimize the number of evaluations of the slow force, with the assumption that the "oscillate" step is cheaper, or comparable in cost, to a single evaluation of the slow force. In our tests, we have used the Verlet scheme for the "oscillate" step within Mollify, with inner stepsize equal to $\varepsilon/100$.

2.3.1 Modified FPU

The Fermi-Pasta-Ulam (FPU) chain is a commonly-used test case for highly oscillatory integrators [10, Sec. XIII.2.1]. The chain is a collection of alternating stiff, harmonic springs and soft, nonlinear springs (see Fig. (1)). After a change of coordinates, the potential can be written as in (6), with Ω constant.
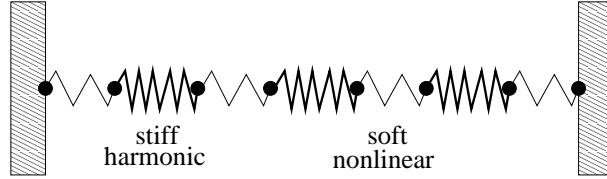

Figure 1: Fermi-Pasta-Ulam spring chain.

We choose $f = s = 3$, corresponding to 3 stiff springs and 4 soft springs (there is one less degree of freedom since the total chain length is prescribed). For positions $\check{q}_1 = (\check{q}_{1,1}, \check{q}_{1,2}, \check{q}_{1,3}) \in \mathbb{R}^3$ and $\check{q}_2 = (\check{q}_{2,1}, \check{q}_{2,2}, \check{q}_{2,3}) \in \mathbb{R}^3$, the potential $\check{V}$ is given by

$$\check{V}(\check{q}_1, \check{q}_2) = \frac{1}{4} \left((\check{q}_{1,1} - \check{q}_{2,1})^4 + \sum_{i=1}^{2} (\check{q}_{1,i+1} - \check{q}_{2,i+1} - \check{q}_{1,i} - \check{q}_{2,i})^4 + (\check{q}_{1,3} + \check{q}_{2,3})^4 \right).$$

We modify the FPU problem to have *non-constant fast frequencies*. We consider a Hamiltonian of the form (6), choose

$$\Omega(\check{q}_1) = \sqrt{1 + \check{q}_{1,1}^2},$$

and use $\check{V}$ above for the slow potential.

The behavior of the modified FPU is qualitatively similar to the original. The fast and slow variables have timescales of $O(\varepsilon)$ and $O(1)$, respectively. In addition there is a slow exchange of the fast actions

$$I_j = \frac{1}{2\Omega(\check{q}_1)} \left[\check{p}_{2,j}^2 + \frac{\Omega(\check{q}_1)^2}{\varepsilon^2} \check{q}_{2,j}^2 \right], \qquad j = 1, 2, 3, \tag{33}$$

13

over long time periods of $O(\varepsilon^{-1})$. The quantity

$$I = I_1 + I_2 + I_3 \tag{34}$$

is an adiabatic invariant (see [1, 2] and [10, Theorem XIII.6.3]): it is an $O(1)$ quantity that is almost preserved over long times, with oscillations only of magnitude $O(\varepsilon)$.

For all numerical experiments, the initial conditions are

$$\check{q}_1 = (1,0,0)^T, \qquad \check{p}_1 = (1,0,0)^T, \qquad \check{q}_2 = (\varepsilon,0,0)^T, \qquad \check{p}_2 = (1,0,0)^T.$$

For these values, we have $H_\varepsilon(0) = 2.5 + 3\varepsilon^2 + 0.5\varepsilon^4 \approx 2.5$ and $I(0) = \dfrac{1}{2}\left(\dfrac{1}{\sqrt{2}} + \sqrt{2}\right) \approx 1.06$.

2.3.2 Preservation of invariants and exchange of actions

We first monitor how the energy (9) and the adiabatic invariant (34) are preserved along the numerical trajectory. Fig. 2 shows the lack of drift in both quantities over long times, for $h = 0.005$, $\varepsilon = 10^{-3}$ and the above choice of initial conditions. Note that energy preservation for symplectic schemes is typically proven in the limit $h \to 0$, for fixed ε. In our case, $h/\varepsilon = 5$, so that energy preservation is not guaranteed by the theory.
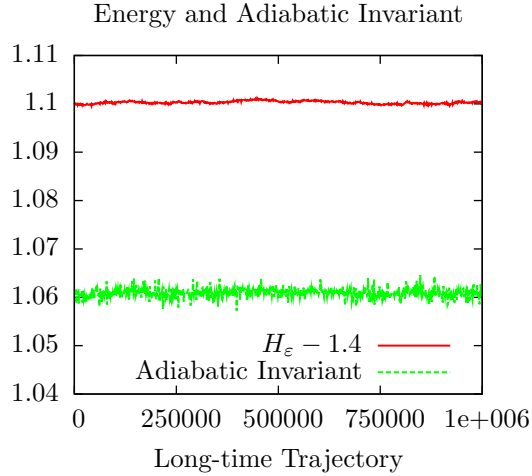

Figure 2: Energy (we plot $H_\varepsilon - 1.4$ to allow better scaling) and adiabatic invariant computed with Algorithm 1 over the long time interval $[0, 10^6]$ for $\varepsilon = 10^{-3}$. We use stepsize $h = 0.005$.

In Fig. 3, we examine the slow exchange among the actions I_j defined in (33). We observe that Algorithm 1 accurately reproduces exchange of the actions, in contrast to Mollify. Note that we have used a smaller stepsize h for Mollify than for Algorithm 1 in order to balance one possible measure of computational cost – the number of evaluations of the slow force. We discuss the issue of computational efficiency in Section 2.3.3.

2.3.3 Resonance and computational efficiency

One common difficulty encountered among integrators for highly oscillatory systems is the appearance of resonances which destroy the long-time preservation of energy and other invariants. In linear cases, these typically occur when the slow time step h is an integer multiple of half the period of the fast motion. In Fig. 4, we explore the resonance behavior by simulating many trajectories, with various
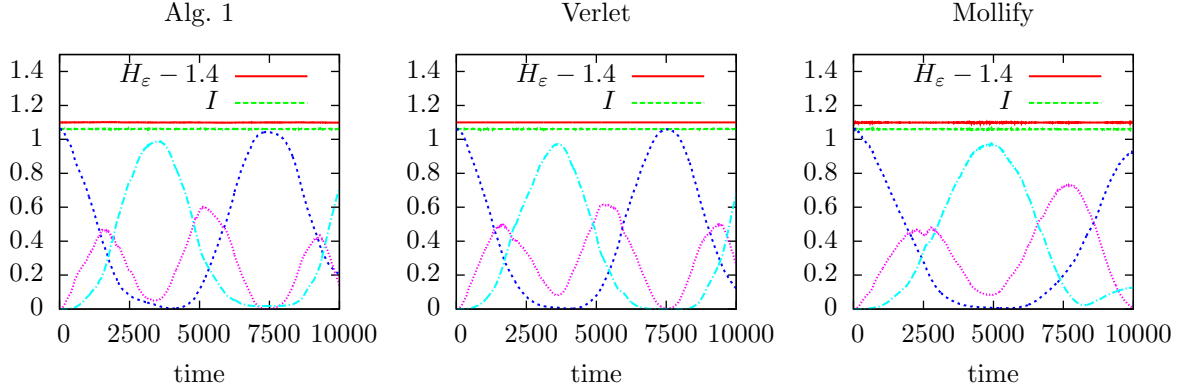
Figure 3: A single trajectory is simulated using three algorithms, Algorithm 1, Verlet, and Mollify. The energy, adiabatic invariant $I = I_1 + I_2 + I_3$, and individual actions I_j are plotted to time 10^4, with parameter $\varepsilon = 10^{-3}$. Verlet is run with a very small time step $h = 10^{-5}$ and is considered as the reference solution. Mollify and Algorithm 1 are run with stepsizes $h = 0.0008$ and $h = 0.02$, respectively, which are chosen so that both algorithms involve roughly the same number of calls to the slow forces. This plot shows that Algorithm 1 better captures the slow exchange of actions for a comparable computational effort.

choices of the ratio h/ε. We do this in two different ways: by holding ε fixed while varying h as well as by holding h fixed while varying ε.

We display the maximum error in energy and maximum variance in I to time $T = 10^4$,

$$\text{err} = \max_{t \in [0,10^4]} |H_\varepsilon(t) - H_\varepsilon(0)|, \qquad \text{var} = \max_{t \in [0,10^4]} |I(t) - I(0)|. \tag{35}$$

Recall that $H_\varepsilon(0) \approx 2.5$ and $I(0) \approx 1.06$. Since the exact trajectory preserves energy, we desire that the method has small error in H_ε. On the other hand, there is variation in $I(t)$ even for the exact trajectory and correctly predicting the variation is also of interest. Therefore, for each ε, we have also computed a reference variation in I. Results are shown in Fig. 4.

For fixed $\varepsilon = 10^{-3}$ (the upper row of Fig. 4), Algorithm 1 performs quite well in the whole range of h, with only a few spikes in the error. Mollify only performs well for small h, with large, generalized resonant regions. For fixed $h = 0.02$ (the lower row of Fig. 4), Algorithm 1 performs better for smaller ε, and the error in the energy blows up for increasing ε. The prediction of the variation in I is even better, matching the reference result. Note that the fact that Algorithm 1 performs better for smaller ε is consistent with the fact that it is derived using homogenization techniques which are valid in the limit $\varepsilon \to 0$. For large ε, the terms neglected in the generating function are no longer small.

We next consider the maximum error in energy versus the computational cost. For the sake of our comparison here, we use the number of evaluations of the slow force $-\nabla \check{V}$ as a proxy for overall computational cost. We thus assume that the slow forces are much more expensive to compute than the fast forces. For example, in a simulation of molecular chains, if the fast forces represent the bonds between adjacent particles in the chain and the slow forces include all other intramolecular as well as long-range intermolecular forces, then the slow force is much more computationally involved. For the Mollify algorithm, this will mean assuming that the computational expense of the 'oscillate' step is negligible in comparison to evaluating the slow forces, which may, depending on the application, underestimate the computational cost. The 'oscillate' step involves propagating the position and momentum according to the fast forces using a small time step and, in addition, propagating the matrix $\begin{bmatrix} Q_q & Q_p \\ P_q & P_p \end{bmatrix}$ of partial derivatives of the position and momentum with respect to the initial
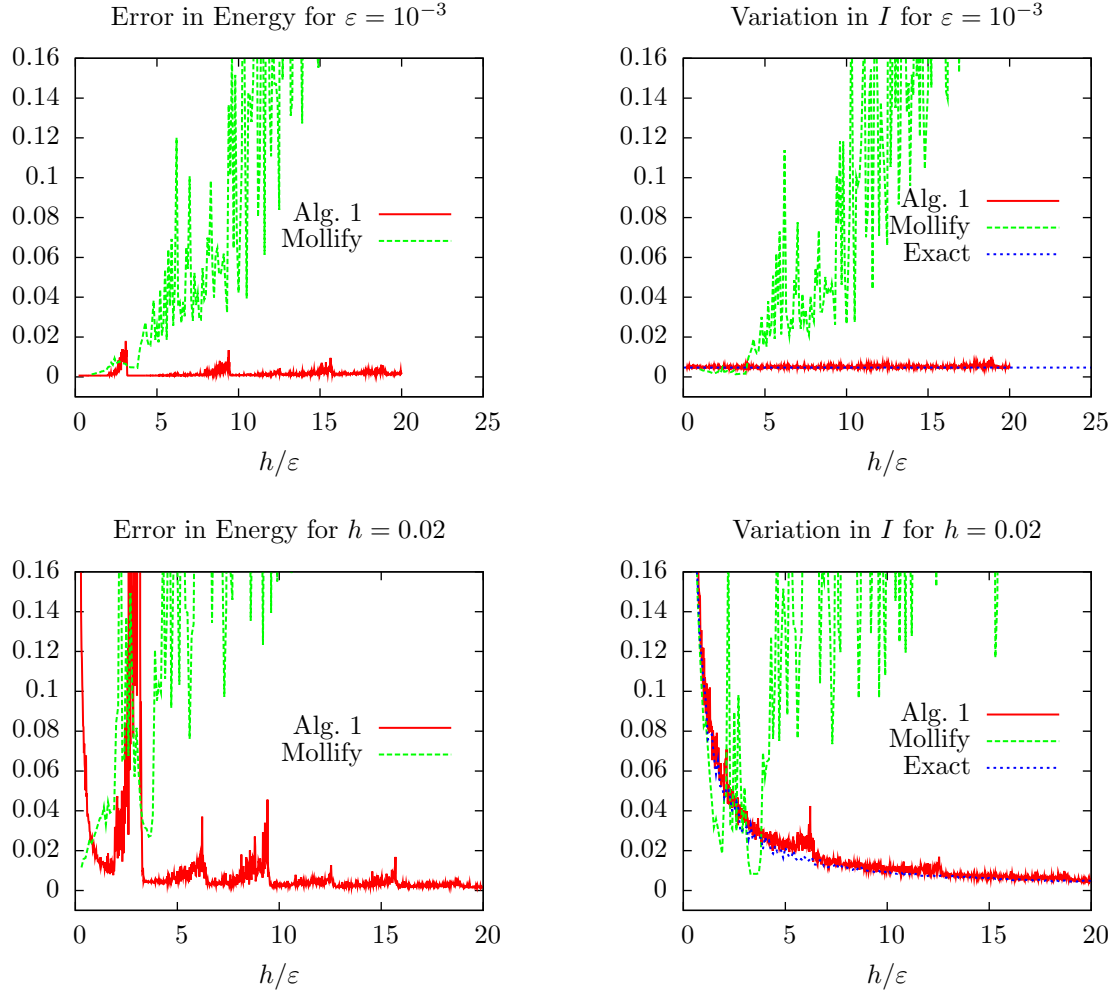
15

Figure 4: Comparison of resonances for Algorithm 1 and Mollify. In the upper row, a series of trajectories are simulated for different stepsizes h, with $\varepsilon = 10^{-3}$. In the lower row, different values of ε are considered, and the dynamics is integrated using the stepsize $h = 0.02$. In both cases, the trajectory is simulated to time $T = 10^4$. In the left (resp. right) column, the maximum error in the energy (resp. the maximum variation in I) over the trajectory, as defined in (35), is plotted. For the variation of I, a reference value is calculated using the Verlet algorithm with a very small time step. Here, Mollify exhibits many more resonances than it does in the case of constant Ω, as shown in [10, Chap. XIII] and [12, Figs. 4-10].
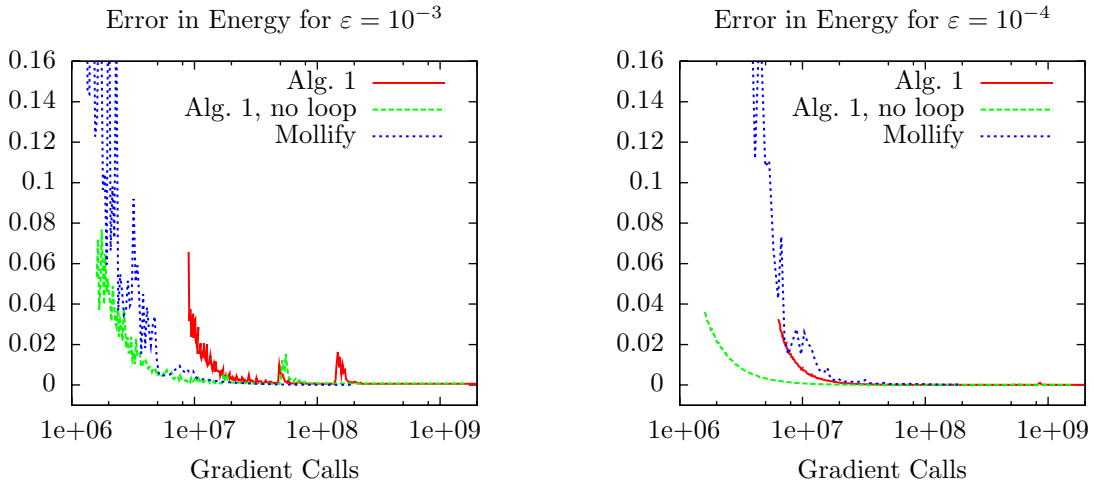
Figure 5: Efficiency plots comparing Algorithm 1, a no-loop variant, and Mollify. For each graph, a series of trajectories are simulated for various stepsizes h with a single set of initial conditions and a single choice of parameter ε ($\varepsilon = 10^{-3}$ for the left plot and $\varepsilon = 10^{-4}$ for the right plot). The trajectory is simulated to time $T = 10^4$, and the maximum variation in the energy over the trajectory is plotted.

condition [7, 13]. This matrix of size $(s + f) \times (s + f)$ is propagated according the Hessian of the fast potential in the so-called variational equation. In practice, simulating the variational equation is much more expensive than simulating a Hamiltonian dynamics on $q \in \mathbb{R}^{s+f}$ according to the fast forces. In the sequel, the cost of the 'oscillate' step in Mollify is entirely neglected.

Figure 5 displays the maximum error in energy to time $T = 10^4$ versus the computational cost. Two plots are shown: we consider both cases $\varepsilon = 10^{-3}$ and $\varepsilon = 10^{-4}$, and integrate the dynamics with different stepsizes h. As the stepsize decreases, the computational cost increases. For all algorithms, the error is generally decreasing as a function of computational cost, up to the presence of resonances. In addition to Algorithm 1 and Mollify, we also consider a variant of Algorithm 1, that does not fully loop in order to solve the implicit equations (31) (see (32)).

Although Algorithm 1 is much more stable with respect to resonances (as shown in Fig. 4), the trade-off is that the computational cost per time-step is increased. Thus, for $\varepsilon = 10^{-3}$, Mollify is slightly cheaper than Algorithm 1 in terms of cost required to achieve a certain accuracy. However, the computational cost of the no-loop variant is much smaller than that of Algorithm 1 and is also smaller than that of Mollify. In addition, the no-loop variant does not exhibit significantly more resonances than does Algorithm 1. For $\varepsilon = 10^{-4}$, Algorithm 1 enjoys a slight edge in comparison to Mollify, and the no-loop variant offers a significant advantage in computational cost.

We also observe from the comparison of the cases $\varepsilon = 10^{-3}$ and $\varepsilon = 10^{-4}$ that, when ε decreases and h is kept fixed, the computational cost of Mollify remains the same, while its accuracy decreases. On the other hand, the cost of Algorithm 1 decreases somewhat since the number of iterations to solve the implicit equations (31) decreases, and the accuracy increases because the terms neglected in the expansion become smaller. In addition, resonances seem to disappear.

3 The case of a matrix-valued, constant frequency

In this section, we consider the case where the fast frequency is a constant matrix. We consider the Hamiltonian (see (7))

$$H_\varepsilon(q_1, q_2, p_1, p_2) = \frac{p_1^T p_1}{2} + \frac{p_2^T p_2}{2} + V(q_1, q_2) + \frac{q_2^T \Omega^2 q_2}{2\varepsilon^2}, \tag{36}$$

and assume that Ω is a *diagonal, positive definite* constant matrix of size $f \times f$ (if Ω is a constant positive definite symmetric matrix, then, upon diagonalizing Ω, we recover the case considered here). Our aim is to show that, upon a slight modification, we can also apply our strategy, thereby extending our previous study in [12] which was restricted to the scalar case $\Omega = \omega \mathrm{Id}$. In the sequel, we have to distinguish two cases, whether or not the eigenvalues of Ω are resonant.

In both cases, we follow the exact same strategy as that we used in [12] and in Section 2: we first precondition the fast motion using a change of variables and next apply our two-scale ansatz to the Hamilton-Jacobi equation associated to the new Hamiltonian. The difference between the non-resonant case and the resonant case lies in the ansatz we make. In the non-resonant case, we introduce a fast time for each frequency in the generating function (see (42)). In contrast, in the resonant case, we introduce a unique fast time for each group of frequencies that are resonant one with each other (see (56)). In both cases, the identification process follows the same lines. The derivation of the scheme is presented in Sections 3.1 and 3.3, in the non-resonant case and in the resonant case, respectively. Numerical results illustrating the non-resonant case are reported in Section 3.2. We conclude this section by considering a test-case with three distinct frequencies, two of which are resonant with each other (see Section 3.4).

3.1 The non-resonant case: derivation of the scheme

Without loss of generality, we assume that the matrix Ω in (36) reads

$$\Omega = \begin{pmatrix} \omega_1 \mathrm{Id}_{m_1} & 0 & \ldots & 0 \\ 0 & \omega_2 \mathrm{Id}_{m_2} & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & \omega_d \mathrm{Id}_{m_d} \end{pmatrix}, \tag{37}$$

with $0 < \omega_1 < \ldots < \omega_d$. The multiplicity of ω_i is m_i, with $\sum_{i=1}^d m_i = f$. In view of the block diagonal decomposition of Ω, we decompose $q_2 \in \mathbb{R}^f$ into d vectors $q_{2,i} \in \mathbb{R}^{m_i}$, such that $q_2 = (q_{2,1}, \ldots, q_{2,d})$ and

$$q_2^T \Omega^2 q_2 = \sum_{i=1}^d \omega_i^2 \, q_{2,i}^T \, q_{2,i}.$$

Likewise, we write $p_2 = (p_{2,1}, \ldots, p_{2,d})$. We assume in the following that the ω_i are non-resonant, in the sense that, for any $k \in \mathbb{Z}^d$,

$$\sum_{j=1}^d \omega_j k_j = 0 \implies k = 0. \tag{38}$$

As pointed out above, we follow here the exact same strategy as that we used in [12], or in Section 2: we first precondition the fast motion, and next consider the Hamilton-Jacobi form of the equations. We proceed first with the time-dependent change of variables $(q_{2,i}, p_{2,i}) \in \mathbb{R}^{2m_i} \mapsto (x_{2,i}, y_{2,i}) = \chi_i(t, q_{2,i}, p_{2,i}) \in \mathbb{R}^{2m_i}$ defined by

$$q_{2,i} = \cos\left(\frac{\omega_i t}{\varepsilon}\right) x_{2,i} + \frac{\varepsilon}{\omega_i} \sin\left(\frac{\omega_i t}{\varepsilon}\right) y_{2,i}, \quad p_{2,i} = -\frac{\omega_i}{\varepsilon} \sin\left(\frac{\omega_i t}{\varepsilon}\right) x_{2,i} + \cos\left(\frac{\omega_i t}{\varepsilon}\right) y_{2,i},$$

which reads, in a more compact form,

$$q_2 = \cos\left(\frac{\Omega t}{\varepsilon}\right) x_2 + \varepsilon \Omega^{-1} \sin\left(\frac{\Omega t}{\varepsilon}\right) y_2, \quad p_2 = -\frac{\Omega}{\varepsilon} \sin\left(\frac{\Omega t}{\varepsilon}\right) x_2 + \cos\left(\frac{\Omega t}{\varepsilon}\right) y_2, \tag{39}$$

with $x_2 = (x_{2,1}, \ldots, x_{2,d})$ and $y_2 = (y_{2,1}, \ldots, y_{2,d})$. The dynamics on (x_2, y_2) reads

$$\dot{x}_2 = \varepsilon \Omega^{-1} \sin\left(\frac{\Omega t}{\varepsilon}\right) \partial_2 V \left[q_1(t), \cos\left(\frac{\Omega t}{\varepsilon}\right) x_2(t) + \varepsilon \Omega^{-1} \sin\left(\frac{\Omega t}{\varepsilon}\right) y_2(t)\right],$$

$$\dot{y}_2 = -\cos\left(\frac{\Omega t}{\varepsilon}\right) \partial_2 V \left[q_1(t), \cos\left(\frac{\Omega t}{\varepsilon}\right) x_2(t) + \varepsilon \Omega^{-1} \sin\left(\frac{\Omega t}{\varepsilon}\right) y_2(t)\right],$$

and the dynamics on (q_1, x_2, p_1, y_2) is a Hamiltonian dynamics with the time-dependent Hamiltonian

$$H_\varepsilon^{\text{non-reso}}(t, q_1, x_2, p_1, y_2) = \frac{p_1^T p_1}{2} + W_\varepsilon^{\text{non-reso}}\left(\frac{\omega_1 t}{\varepsilon}, \dots, \frac{\omega_d t}{\varepsilon}, q_1, x_2, y_2\right), \qquad (40)$$

where

$$W_\varepsilon^{\text{non-reso}}(\tau_1, \dots, \tau_d, q_1, x_2, y_2) = V\left[q_1, (\cos\tau_1)x_{2,1} + \frac{\varepsilon}{\omega_1}(\sin\tau_1)y_{2,1}, \dots, (\cos\tau_d)x_{2,d} + \frac{\varepsilon}{\omega_d}(\sin\tau_d)y_{2,d}\right].$$

We take the Hamiltonian (40) as a starting point for our manipulations. In the sequel, we proceed with the construction of an approximation $\widetilde{S}_\varepsilon(h, q_1, x_2, P_1, Y_2)$ of the solution $\overline{S}_\varepsilon(h, q_1, x_2, P_1, Y_2)$ to the Hamilton-Jacobi equation associated to (40), for small times h. Observing that the variable x_2 is of order ε, we first perform a change of variables and of unknown function:

$$r_2 = \frac{\Omega}{\varepsilon}x_2 \quad \text{and} \quad S_\varepsilon(t, q_1, r_2, P_1, Y_2) = \overline{S}_\varepsilon\left(t, q_1, \varepsilon\Omega^{-1}r_2, P_1, Y_2\right),$$

so that S_ε satisfies

$$\begin{cases} \partial_t S_\varepsilon = \dfrac{P_1^T P_1}{2} + W_\varepsilon^{\text{non-reso}}\left(\dfrac{\omega_1 t}{\varepsilon}, \dots, \dfrac{\omega_d t}{\varepsilon}, q_1 + \partial_{P_1}S_\varepsilon, \varepsilon\Omega^{-1}r_2 + \partial_{Y_2}S_\varepsilon, Y_2\right), \\ S_\varepsilon(0, q_1, r_2, P_1, Y_2) = 0. \end{cases} \qquad (41)$$

We make the ansatz

$$\begin{aligned} S_\varepsilon(t, q_1, r_2, P_1, Y_2) &= S_0\left(t, \tau_1, \dots, \tau_d, q_1, r_2, P_1, Y_2\right) + \varepsilon S_1\left(t, \tau_1, \dots, \tau_d, q_1, r_2, P_1, Y_2\right) \qquad (42) \\ &+ \text{higher order terms in } \varepsilon^k, \ k \geq 2, \end{aligned}$$

where the fast times τ_i are defined by

$$\tau_i = \frac{t\omega_i}{\varepsilon}, \quad 1 \leq i \leq d, \qquad (43)$$

and where the functions $(S_k)_{k\geq 0}$ are assumed to be 2π periodic with respect to each τ_i. We also assume that the functions S_k are of class $C^{1+\alpha}$ with respect to each τ_i, with $\alpha \in \mathbb{N}$, $\alpha > d$.

Remark 3.1. In the case where V does not depend on q_2, the solution to (41) can be analytically identified and is indeed of the form (42).

We now insert (42) in (41), identify the first d variables of $W_\varepsilon^{\text{non-reso}}$ with the fast times $\{\tau_i\}_{1\leq i\leq d}$, and expand in powers of ε. Based on (5), we have that $X_2 = x_2 + \partial_{Y_2}\overline{S}_\varepsilon$. The fast position X_2 is of order ε, so S_0 does not depend on Y_2. The equation of order ε^{-1} in the expansion of (41) then becomes

$$\sum_{i=1}^d \omega_i \ \partial_{\tau_i}S_0 = 0.$$

Using Lemma 3.1 below, we deduce that

$$\forall 1 \leq i \leq d, \quad \partial_{\tau_i}S_0 = 0.$$

Lemma 3.1. *Let $f(\tau_1, \dots, \tau_d)$ be a function that is 2π periodic with respect to each τ_j, and of class $C^{1+\alpha}$ with respect to each of its argument, with $\alpha \in \mathbb{N}$, $\alpha > d$. Assume that*

$$\sum_{j=1}^d \omega_j \ \partial_{\tau_j}f = c, \qquad (44)$$

for a constant c, and a d-tuple $\{\omega_j\}_{1\leq j\leq d}$ satisfying the non-resonance condition (38). Then the function f is a constant and $c = 0$.

Proof. As f is periodic, we can write it as its Fourier series: denoting $\tau = (\tau_1, \ldots, \tau_d)$, we have

$$f(\tau) = \sum_{(k_1, \ldots, k_d) \in \mathbb{Z}^d} f_k \exp(ik \cdot \tau).$$

The assumption (44) reads

$$i \sum_{j=1}^{d} \sum_{(k_1, \ldots, k_d) \in \mathbb{Z}^d} \omega_j f_k k_j \exp(ik \cdot \tau) = c.$$

As f is of class $C^{1+\alpha}$, we have that $|f_k| \leq C(1 + |k|)^{-1-\alpha}$ for some constant C, where we recall $|k| = \sum_{j=1}^{d} |k_j|$. Hence, we have that $\sum_{k \in \mathbb{Z}^d} |f_k k_j| < \infty$, and the above sum is well-defined. We deduce that

$$\forall k \in \mathbb{Z}^d, \quad k \neq 0, \quad f_k \sum_{j=1}^{d} \omega_j k_j = 0, \tag{45}$$

whereas the identification for $k = 0$ leads to $c = 0$. We infer from (45) and assumption (38) that $f_k = 0$ for any $k \in \mathbb{Z}^d$, $k \neq 0$. So $f(\tau) = f_0$. This concludes the proof. $\qquad\square$

The equation of order ε^0 reads

$$\partial_t S_0 + \sum_{i=1}^{d} \omega_i \partial_{\tau_i} S_1 = \frac{P_1^T P_1}{2} + V(q_1 + \partial_{P_1} S_0, 0). \tag{46}$$

Since S_0 does not depend on τ and S_1 is 2π periodic with respect to each τ_i, we can again apply Lemma 3.1. We thus infer from (46) that

$$\partial_t S_0 = \frac{P_1^T P_1}{2} + V(q_1 + \partial_{P_1} S_0, 0) \tag{47}$$

and $\partial_{\tau_i} S_1 = 0$ for all $1 \leq i \leq d$. Equation (47) is supplied with the initial condition $S_0(t = 0, q_1, r_2, P_1) = 0$. For each r_2, we thus recognize the Hamilton-Jacobi equation for the Hamiltonian function

$$H_1(q_1, p_1) = \frac{p_1^T p_1}{2} + V(q_1, 0). \tag{48}$$

So S_0 does not depend on r_2. In the sequel, we will approximate $S_0(t, q_1, P_1)$ by

$$S_0^{\mathrm{SE}}(t, q_1, P_1) = S_0(0, q_1, P_1) + t\partial_t S_0(0, q_1, P_1) = t\left(\frac{P_1^T P_1}{2} + V(q_1, 0)\right), \tag{49}$$

which amounts to integrating the Hamiltonian dynamics generated by (48) with the symplectic Euler algorithm. We have $S_0(t) = S_0^{\mathrm{SE}}(t) + O(t^2)$.

The sequel of the identification is not difficult. The bottom line is again as follows: since the $\{\tau_i\}$ are independent variables in each S_k, we can integrate with respect to each one to split the equations and close the hierarchy. Following arguments similar to those presented in [12], we find that

$$S_1 \equiv 0 \tag{50}$$

20

and that $S_2(t) = S_2^{\text{SE}}(t) + O(t^2)$, with

$$
\begin{aligned}
S_2^{\text{SE}}(t, \tau_1, \ldots, \tau_d, q_1, r_2, P_1, Y_2) \;=\;& \sum_{i=1}^{d} \frac{1}{\omega_i^2} (\nabla_{2,i} V)^T Y_{2,i} \\
& +\; \sum_{i=1}^{d} \frac{1}{\omega_i^2} (\nabla_{2,i} V(q_1 + tP_1, 0))^T \left((\sin \tau_i) r_{2,i} - (\cos \tau_i) Y_{2,i} \right) \\
& -\; \frac{t}{2} \sum_{i=1}^{d} \frac{1}{\omega_i^2} (\nabla_{2,i} V)^T \nabla_{2,i} V \\
& +\; \frac{t}{4} \sum_{i=1}^{d} \frac{1}{\omega_i^2} \left(r_{2,i}^T \nabla_{2,i}^2 V r_{2,i} + Y_{2,i}^T \nabla_{2,i}^2 V Y_{2,i} \right),
\end{aligned}
\tag{51}
$$

where the derivatives of V are, unless otherwise mentioned, evaluated at $(q_1, 0)$, and where $\nabla_{2,i}^2 V$ is the Hessian matrix of V with respect to $q_{2,i}$.

Observe that, in (51), there is no term coupling components associated to different frequencies ω_i and ω_j, $j \neq i$. This is reminiscent of the fact that, in the ansatz, the fast times τ_i are independent variables, and that each S_k is 2π periodic with respect to *each* τ_i.

Consider now the approximation $S_\varepsilon(h) \approx S_\varepsilon^{\text{non}-\text{reso}}(h)$, with

$$
S_\varepsilon^{\text{non}-\text{reso}}(h) := S_0^{\text{SE}}(h) + \varepsilon S_1(h) + \varepsilon^2 S_2^{\text{SE}}(h),
$$

where S_0^{SE}, S_1 and S_2^{SE} are respectively defined by (49), (50) and (51). Using this approximation, we obtain a symplectic algorithm in the variables (q_1, x_2, p_1, y_2). Returning to the original variables (q_1, q_2, p_1, p_2), we obtain the symplectic Algorithm 2 outlined below, which we denote by $(Q_1, Q_2, P_1, P_2) = \Psi_h^{\text{non}-\text{reso1}}(q_1, q_2, p_1, p_2)$.

Algorithm 2 (Preconditioned Symplectic Scheme $\Psi_h^{\mathrm{non-reso1}}(q_1,q_2,p_1,p_2)$). Set $(q_1,q_2,p_1,p_2) = (q_1^n,q_2^n,p_1^n,p_2^n)$, $\tau_i = \omega_i h/\varepsilon$ and perform the following steps:

1. Change of variables: set $\quad x_2 = q_2, \quad y_2 = p_2, \quad r_2 = \Omega x_2/\varepsilon$.

2. Solve for (P_1, Y_2) in the equations

$$
\begin{cases}
y_2 &=& Y_2 + \dfrac{h\varepsilon}{2}\displaystyle\sum_{i=1}^{d}\dfrac{1}{\omega_i}\nabla_{2,i}^2 V(q_1,0)r_{2,i} + \varepsilon\sum_{i=1}^{d}\dfrac{\sin\tau_i}{\omega_i}\nabla_{2,i}V(q_1+hP_1,0), \\[4mm]
p_1 &=& P_1 + h\nabla_1 V(q_1,0) + \varepsilon^2\displaystyle\sum_{i=1}^{d}\dfrac{1}{\omega_i^2}\nabla_{12i}V(q_1,0)Y_{2,i} \\[4mm]
&+& \varepsilon^2\displaystyle\sum_{i=1}^{d}\dfrac{1}{\omega_i^2}\nabla_{12i}V(q_1+hP_1,0)\left((\sin\tau_i)r_{2,i} - (\cos\tau_i)Y_{2,i}\right) \\[4mm]
&-& h\varepsilon^2\displaystyle\sum_{i=1}^{d}\dfrac{1}{\omega_i^2}\nabla_{12i}V(q_1,0)\nabla_{2,i}V(q_1,0) \\[4mm]
&+& \dfrac{h\varepsilon^2}{4}\displaystyle\sum_{i=1}^{d}\dfrac{1}{\omega_i^2}(r_{2,i}^T\nabla_{12i2i}V(q_1,0)r_{2,i} + Y_{2,i}^T\nabla_{12i2i}V(q_1,0)Y_{2,i}).
\end{cases}
$$

3. Set $\quad Q_1 = q_1 + hP_1 + h\varepsilon^2\displaystyle\sum_{i=1}^{d}\dfrac{1}{\omega_i^2}\nabla_{12i}V(q_1+hP_1,0)\left((\sin\tau_i)r_{2,i} - (\cos\tau_i)Y_{2,i}\right).$

4. Set

$$
X_2 = x_2 + \varepsilon^2\sum_{i=1}^{d}\frac{1}{\omega_i^2}\nabla_{2,i}V(q_1,0) + h\varepsilon^2\sum_{i=1}^{d}\frac{1}{2\omega_i^2}\nabla_{2,i}^2 V(q_1,0)Y_{2,i} - \varepsilon^2\sum_{i=1}^{d}\frac{\cos\tau_i}{\omega_i^2}\nabla_{2,i}V(q_1+hP_1,0).
$$

5. Return to the original variables: set $\tau = \Omega h/\varepsilon$ and

$$
Q_2 = (\cos\tau)X_2 + \varepsilon\Omega^{-1}(\sin\tau)Y_2, \quad P_2 = -\frac{\Omega}{\varepsilon}(\sin\tau)X_2 + (\cos\tau)Y_2.
$$

Set $\left(q_1^{n+1}, q_2^{n+1}, p_1^{n+1}, p_2^{n+1}\right) = (Q_1, Q_2, P_1, P_2)$.

Note that, at step 2, we need to solve a system implicit in $Z = (P_1, Y_2)$, which reads $z = Z + hF(Z) + \varepsilon G(Z)$ with $z = (p_1, y_2)$. In practice, we use a fixed point method, which converges in only a few iterations (three iterations in the test-case considered below).

Remark 3.2. High-order derivatives of V appear in Algorithm 2. As shown in Section 2.2 above, they can be replaced by a finite difference approximation in the generating function.

Neglecting all terms of order ε^3, the scheme $\Psi_h^{\mathrm{non-reso1}}(q_1,q_2,p_1,p_2)$ is first order in h. A simple, well-known, manner to get a scheme of higher order is to consider the symmetric form

$$
(Q_1, Q_2, P_1, P_2) = \Psi_h^{\mathrm{non-reso2}}(q_1,q_2,p_1,p_2) = \left(\Psi_{h/2}^{\mathrm{non-reso1}}\right)^* \Psi_{h/2}^{\mathrm{non-reso1}}(q_1,q_2,p_1,p_2),
$$

where Ψ^* denotes the adjoint of Ψ. This scheme, denoted Algorithm 3 in the sequel, is symplectic, symmetric and, neglecting all terms of order ε^3, of order 2 in h.

> *Algorithm* 3 (Preconditioned Symplectic Scheme $\Psi_h^{\text{non}-\text{reso2}}(q_1, q_2, p_1, p_2)$). Set $(q_1, q_2, p_1, p_2) = (q_1^n, q_2^n, p_1^n, p_2^n)$ and perform the following steps:
>
> 1. Set $(\overline{Q}_1, \overline{Q}_2, \overline{P}_1, \overline{P}_2) = \Psi_{h/2}^{\text{non}-\text{reso1}}(q_1, q_2, p_1, p_2)$.
>
> 2. Set $(Q_1, Q_2, P_1, P_2) = \left(\Psi_{h/2}^{\text{non}-\text{reso1}}\right)^* (\overline{Q}_1, \overline{Q}_2, \overline{P}_1, \overline{P}_2)$.
>
> Set $\left(q_1^{n+1}, q_2^{n+1}, p_1^{n+1}, p_2^{n+1}\right) = (Q_1, Q_2, P_1, P_2)$.

3.2 The non-resonant case: numerical results

We consider a Hamiltonian of the form (36), with $q_1 \in \mathbb{R}$ and $q_2 = (q_{2,1}, q_{2,2}, q_{2,3}) \in \mathbb{R}^3$, and where the slow potential energy is

$$V(q_1, q_2) = (c + q_{2,1} + q_{2,2} + \gamma q_{2,3})^4 + \frac{1}{8}q_1^2 q_{2,1}^2 + \frac{1}{2}q_1^2$$

with $c = 1$ and $\gamma = 2.5$. We choose $\Omega = \text{diag}\left(1, 1, \sqrt{2}\right)$ as the matrix of fast frequencies.

Let I_j denote the energy associated to each fast degree of freedom:

$$I_j = \frac{p_{2,j}^2}{2} + \frac{\overline{\omega}_j^2 \, q_{2,j}^2}{2\varepsilon^2}, \quad 1 \leq j \leq 3,$$

with $\overline{\omega}_1 = \overline{\omega}_2 = 1$ and $\overline{\omega}_3 = \sqrt{2}$. We first note that here we use I_j to denote the fast energy, as opposed to Section 2 where it denotes the fast action (energy divided by frequency). Of course, for the constant frequency case here, these two quantities differ only by a multiplicative constant. In addition, we use here a different convention than that of Section 3.1 on how the eigenvalues of Ω are numbered. It is well-known (see [10, Sec. XIII.9]) that the quantities

$$I = \sum_{j=1}^3 I_j \quad \text{and} \quad I_3 \tag{52}$$

are adiabatic invariants of the dynamics. In contrast, I_1 and I_2, associated to the same frequency, are *not* adiabatic invariants, although their sum, $I_1 + I_2 = I - I_3$, is.

We first choose $\varepsilon = 1/70$ and $h = 10\varepsilon$, and we monitor the evolution of the energy and adiabatic invariants up to time $T = 10^6$ on the numerical trajectory computed with Algorithm 3. Results are shown in Fig. 6. We observe no drift.

For the same parameters, we show in Fig. 7 the evolution of I_j over the time window $[0, 50]$. As expected, I_3 is preserved, as well as $I_1 + I_2$. We observe that Algorithm 3 correctly reproduces the exchange between I_1 and I_2.

We now focus on the robustness of Algorithm 3 as ε decreases. We set the time step to $h = 0.02$ and consider the variations of the energy and of the adiabatic invariants (52),

$$\max_{t \in [0, 10^4]} \frac{|H(t) - H(0)|}{H(0)}, \quad \max_{t \in [0, 10^4]} \frac{|I(t) - I(0)|}{I(0)}, \quad \max_{t \in [0, 10^4]} \frac{|I_3(t) - I_3(0)|}{I_3(0)}, \tag{53}$$

over the time interval $t \in [0, 10^4]$, for stiffness ε varying between 10^{-3} to 1. Results are shown on Fig. 8. Only a few resonances can be seen, and they are extremely peaked. In addition, the algorithm is very stable as ε decreases to 0.
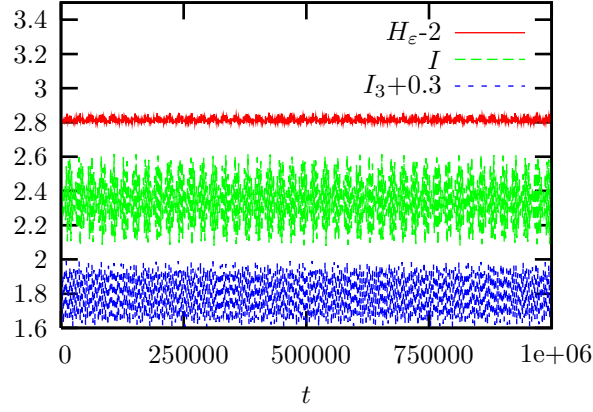
Figure 6: Energy and adiabatic invariants I and I_3 (for convenience, we plot $H_\varepsilon - 2$, I and $I_3 + 0.3$) along the trajectory computed with Algorithm 3 ($\varepsilon = 1/70$ and $h = 10\varepsilon$).
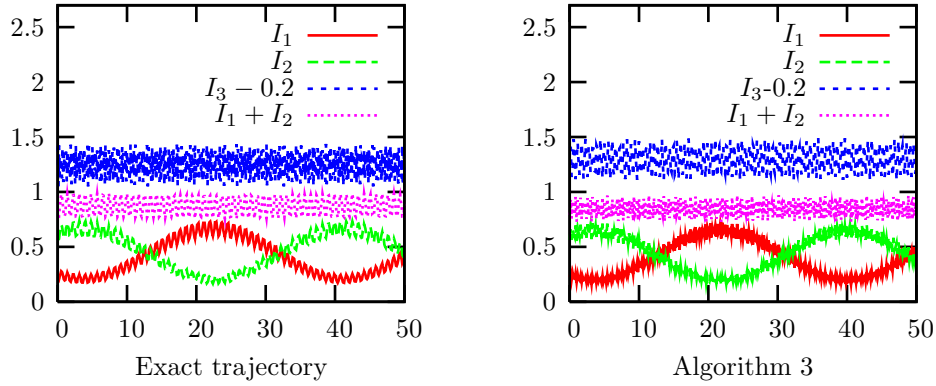

Figure 7: Preservation of the adiabatic invariants $I_1 + I_2$ and I_3, and exchange between I_1 and I_2, for $\varepsilon = 1/70$ (Algorithm 3 has been used with $h = 10\varepsilon$).
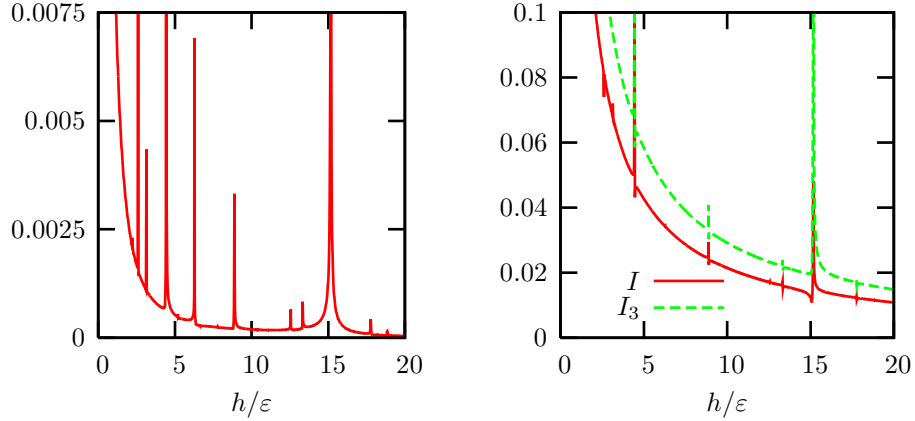

Figure 8: Maximum variations (53) of the energy (left) and of the adiabatic invariants I and I_3 (right) on the time interval $[0, 10^4]$, for several ε ($h = 0.02$), for Algorithm 3.

3.3 The resonant case: derivation of the scheme

In this section, we again consider the Hamiltonian (36), with a diagonal matrix $\Omega \in \mathbb{R}^{f \times f}$. In contrast to Section 3.1, we assume here that some entries of the diagonal of Ω are resonant. To simplify the notation and the analysis, we assume that Ω is of the form (37) with just $d = 2$, namely

$$\Omega = \begin{pmatrix} \omega_a \mathrm{Id}_{m_a} & 0 \\ 0 & \omega_b \mathrm{Id}_{m_b} \end{pmatrix},$$

with $m_a + m_b = f$, $0 < \omega_a < \omega_b$, and

$$\omega_b = \frac{\beta}{\alpha}\omega_a, \quad \text{with } \alpha \in \mathbb{N}^\star, \ \beta \in \mathbb{N}^\star, \ \alpha \neq \beta, \ \text{and } \alpha \text{ and } \beta \text{ coprime.} \tag{54}$$

Likewise, we write $q_2 = (q_{2,a}, q_{2,b})$ with $q_{2,a} \in \mathbb{R}^{m_a}$ and $q_{2,b} \in \mathbb{R}^{m_b}$, and $p_2 = (p_{2,a}, p_{2,b})$.

As in Section 3.1, we first consider the time-dependent change of variables (39), which we write $(x_2, y_2) = \chi(t, q_2, p_2) \in \mathbb{R}^{2f}$. The dynamics on (q_1, x_2, p_1, y_2) is a Hamiltonian dynamics with the time-dependent Hamiltonian

$$H_\varepsilon^{\mathrm{reso}}(t, q_1, x_2, p_1, y_2) = \frac{p_1^T p_1}{2} + W_\varepsilon^{\mathrm{reso}}\left(\frac{\omega_a t}{\alpha \varepsilon}, q_1, x_2, y_2\right),$$

where

$$W_\varepsilon^{\mathrm{reso}}(\tau, q_1, x_2, y_2) = V\left[q_1, (\cos \alpha\tau)x_{2,a} + \frac{\varepsilon}{\omega_a}(\sin \alpha\tau)y_{2,a}, (\cos \beta\tau)x_{2,b} + \frac{\varepsilon}{\omega_b}(\sin \beta\tau)y_{2,b}\right].$$

We let $\overline{S}_\varepsilon(t, q_1, x_2, P_1, Y_2)$ denote the solution to the Hamilton-Jacobi equation associated with $H_\varepsilon^{\mathrm{reso}}$ and perform the change of variables and of unknown function

$$r_2 = \frac{\Omega}{\varepsilon}x_2 \quad \text{and} \quad S_\varepsilon(t, q_1, r_2, P_1, Y_2) = \overline{S}_\varepsilon\left(t, q_1, \varepsilon\Omega^{-1}r_2, P_1, Y_2\right).$$

The function S_ε satisfies

$$\begin{cases} \partial_t S_\varepsilon = \dfrac{P_1^T P_1}{2} + W_\varepsilon^{\mathrm{reso}}\left(\dfrac{\omega_a t}{\alpha \varepsilon}, q_1 + \partial_{P_1}S_\varepsilon, \varepsilon\Omega^{-1}r_2 + \partial_{Y_2}S_\varepsilon, Y_2\right), \\[2mm] S_\varepsilon(0, q_1, r_2, P_1, Y_2) = 0. \end{cases} \tag{55}$$

We make the ansatz

$$\begin{aligned} S_\varepsilon(t, q_1, r_2, P_1, Y_2) &= S_0\left(t, \tau, q_1, r_2, P_1, Y_2\right) + \varepsilon S_1\left(t, \tau, q_1, r_2, P_1, Y_2\right) \\ &+ \text{ higher order terms in } \varepsilon^k, \ k \geq 2, \end{aligned} \tag{56}$$

where the fast time τ is defined by

$$\tau = \frac{t\omega_a}{\alpha \varepsilon} = \frac{t\omega_b}{\beta \varepsilon}, \tag{57}$$

and where the functions $(S_k)_{k \geq 0}$ are assumed to be 2π periodic with respect to τ.

Remark 3.3. In the case where V does not depend on q_2, the solution to (55) can be analytically identified and is indeed of the form (56).

We now insert (56) in (55), identify the first variable of $W_\varepsilon^{\mathrm{reso}}$ with the fast time τ, and expand in powers of ε. As in Section 3.1, S_0 is independent from Y_2 and τ. The equation of order ε^0 reads

$$\partial_t S_0 + \frac{\omega_a}{\alpha}\partial_\tau S_1 = \frac{P_1^T P_1}{2} + V(q_1 + \partial_{P_1}S_0, 0). \tag{58}$$

25

Since S_0 does not depend on τ and S_1 is 2π periodic with respect to τ, we infer from (58) that

$$\partial_t S_0 = \frac{P_1^T P_1}{2} + V(q_1 + \partial_{P_1} S_0, 0) \tag{59}$$

and $\partial_\tau S_1 = 0$. Equation (59) is supplied with the initial condition $S_0(t = 0, q_1, r_2, P_1) = 0$. We again recognize the Hamilton-Jacobi equation for the Hamiltonian function (48). In the sequel, S_0 is thus again approximated by (49).

Following the same arguments as in [12], we next proceed with the sequel of the identification, using the fact that, since $\alpha \neq \beta$ in (54),

$$\int_0^{2\pi} \cos \alpha\tau \ \cos \beta\tau \ d\tau = 0.$$

As a consequence, at least at the orders of the expansion that we consider, no coupling appears between the terms associated to the frequency ω_a and those associated with the frequency ω_b. We find that

$$S_1 \equiv 0 \tag{60}$$

and that $S_2(t) = S_2^{\mathrm{SE}}(t) + O(t^2)$, with

$$
\begin{aligned}
S_2^{\mathrm{SE}}(t, \tau, q_1, r_2, P_1, Y_2) \ &= \ \frac{1}{\omega_a^2}(\nabla_{2,a} V)^T Y_{2,a} + \frac{1}{\omega_b^2}(\nabla_{2,b} V)^T Y_{2,b} \\
&+ \ \frac{1}{\omega_a^2}(\nabla_{2,a} V(q_1 + tP_1, 0))^T \left((\sin \alpha\tau) r_{2,a} - (\cos \alpha\tau) Y_{2,a}\right) \\
&+ \ \frac{1}{\omega_b^2}(\nabla_{2,b} V(q_1 + tP_1, 0))^T \left((\sin \beta\tau) r_{2,b} - (\cos \beta\tau) Y_{2,b}\right) \quad (61) \\
&- \ \frac{t}{2}\left(\frac{1}{\omega_a^2}(\nabla_{2,a} V)^T \nabla_{2,a} V + \frac{1}{\omega_b^2}(\nabla_{2,b} V)^T \nabla_{2,b} V\right) \\
&+ \ \frac{t}{4\omega_a^2}\left(r_{2,a}^T \nabla_{2,a}^2 V r_{2,a} + Y_{2,a}^T \nabla_{2,a}^2 V Y_{2,a}\right) \\
&+ \ \frac{t}{4\omega_b^2}\left(r_{2,b}^T \nabla_{2,b}^2 V r_{2,b} + Y_{2,b}^T \nabla_{2,b}^2 V Y_{2,b}\right),
\end{aligned}
$$

where the derivatives of V are evaluated at $(q_1, 0)$ unless otherwise mentioned, and where $\nabla_{2,a}^2 V$ is the Hessian matrix of V with respect to $q_{2,a}$.

Observe that, in (61), there is no term coupling components associated to different frequencies. This is reminiscent of the fact that, at the $O(\varepsilon^2)$ term that we consider here, coupling terms can only come from terms containing products of the form $\cos \alpha\tau \ \cos \beta\tau$, the average of which vanishes. In contrast, to identify S_3, we need to handle terms of the form $(\cos \alpha\tau)^s \ (\cos \beta\tau)^r$ (with $r + s = 3$), whose average may not vanish (e.g. in the case $\alpha = 1$, $\beta = 2$, $s = 2$ and $r = 1$).

We next observe that the generating functions S_0^{SE}, S_1 and S_2^{SE} that we have identified in this resonant case, defined by (49), (60) and (61), where the fast time τ is given by (57), are equal to the generating functions S_0^{SE}, S_1 and S_2^{SE} identified in the non-resonant case, see (49), (50) and (51), where the fast times τ_i are given by (43). As a consequence, even though the frequencies ω_a and ω_b are here resonant, we again obtain the algorithms 2 and 3 proposed above.

3.4 The resonant case: numerical results

In Sections 3.1 and 3.3, we have derived algorithms to integrate the dynamics in the non-resonant case and in the resonant case, respectively. As explained at the end of the previous section, when

expanding up to $O(\varepsilon^2)$, these two cases turn out to yield the same algorithm. A natural intuition, which is confirmed by a careful identification similar to the ones performed above, is that this algorithm is also appropriate in the case when the matrix of fast frequencies is of the form (37), where the ω_i again satisfy $0 < \omega_1 < \omega_2 < \ldots < \omega_d$, but where, in contrast to the situation considered in Sections 3.1 and 3.3, they are now possibly resonant, and d is arbitrary.

Consider a system of the form (36), with $q_1 \in \mathbb{R}$ and $q_2 = (q_{2,1}, q_{2,2}, q_{2,3}, q_{2,4}) \in \mathbb{R}^4$, and where the slow potential energy is

$$V(q_1, q_2) = (c + q_{2,1} + q_{2,2} + q_{2,3} + \gamma q_{2,4})^4 + \frac{1}{8} q_1^2 q_{2,1}^2 + \frac{1}{2} q_1^2$$

with $c = 1$ and $\gamma = 2.5$. We choose the matrix of fast frequencies to be $\Omega = \mathrm{diag}\left(1, 1, \sqrt{2}, 2\right)$. A very similar test-case has already been studied in [3] and [10, Sec. XIII.9.1], where the chosen value of $c = 1/20$ causes the exchange between I_1 and I_2 to occur on a slower time scale. It obviously enters the framework of the current section, but not the framework of Section 3.1, as the first and the last frequencies are resonant. In this case, the adiabatic invariants are

$$I = \sum_{j=1}^{4} I_j \quad \text{and} \quad I_3, \tag{62}$$

where I_j is the energy associated to the jth fast degree of freedom:

$$I_j = \frac{p_{2,j}^2}{2} + \frac{\overline{\omega}_j^2 \, q_{2,j}^2}{2\varepsilon^2}, \quad 1 \le j \le 4,$$

with $\overline{\omega}_1 = \overline{\omega}_2 = 1$, $\overline{\omega}_3 = \sqrt{2}$ and $\overline{\omega}_4 = 2$.

We first choose $\varepsilon = 1/70$ and $h = 10\varepsilon$, and we monitor the evolution of the energy and adiabatic invariants up to time $T = 10^6$ on the numerical trajectory computed with Algorithm 3. Results are very similar to those shown on Fig. 6: we do not observe any drift.

We now focus on the robustness of Algorithm 3, as ε decreases. We set the time step to $h = 0.02$, and consider the variations (53) of the energy and of the adiabatic invariants (62), over the time interval $t \in [0, 10^4]$, for stiffness ε varying between 10^{-3} to 1. Results are shown on Fig. 9. When ε decreases to 0, the algorithm performs better and better, except for a few peaked resonances.
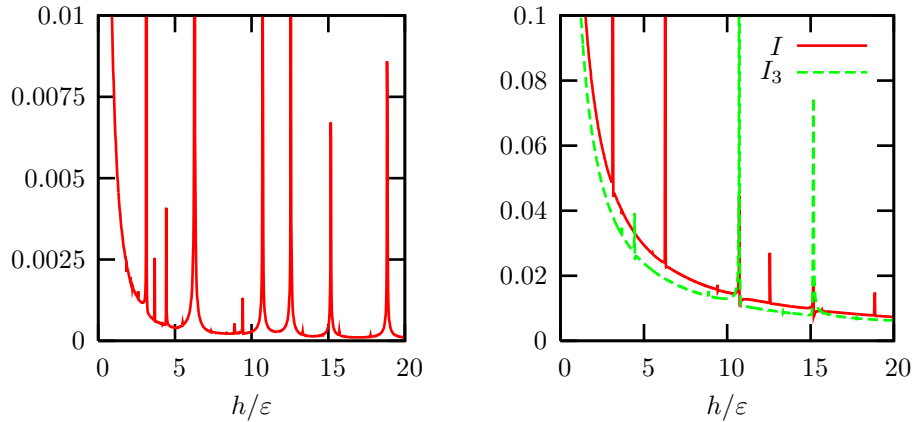

Figure 9: Maximum variations (53) of the energy (left) and of the adiabatic invariants I and I_3 (right) on the time interval $[0, 10^4]$, for several ε ($h = 0.02$), for Algorithm 3. There is an increase in resonances compared to Fig. 8, though as before they are very tightly peaked.

We conclude this section by discussing the exchange of fast energies. In the case at hand, it is well-known (see [10, Sec. XIII.9]) that some exchange occurs between I_1, I_2 (which are both associated to the same frequency $\overline{\omega}_1$) and I_4 (which is associated to a frequency $\overline{\omega}_4$ resonant with $\overline{\omega}_1$). The exchange between I_1 and I_2 is $O(1)$ over timescales of $O(\varepsilon^{-1})$, which is stronger than the exchange between I_4 and $I_1 + I_2$, which is only $O(\varepsilon)$ over timescales of $O(\varepsilon^{-1})$. For some choices of parameters, there is an $O(1)$ exchange between I_4 and $I_1 + I_2$ that occurs on the long time scale $O(\varepsilon^{-2})$. It turns out that, on the numerical trajectory computed using Algorithm 3, the exchange between $I_1 + I_2$ and I_4 on long time scales is not reproduced, and I_4 is almost preserved. This is due to the fact that (i) resonant frequencies are handled by the algorithm we derived as if they were non-resonant (see Section 3.3) and (ii) in the non-resonant case, no exchange occurs between the fast energies. If we were to expand the generating function to third order in ε, certain exchange terms among the fast terms would be included in the resulting algorithm, which could potentially capture the exchange between $I_1 + I_2$ and I_4.

Yet, the fact that we chose not to include these terms did not destroy the preservation of energy and adiabatic invariants, which are accurately recovered, as pointed out above. In addition, the exchange between I_1 and I_2, that occurs on the time scale $O(\varepsilon^{-1})$, is accurately reproduced by Algorithm 3, as shown on Fig. 10.
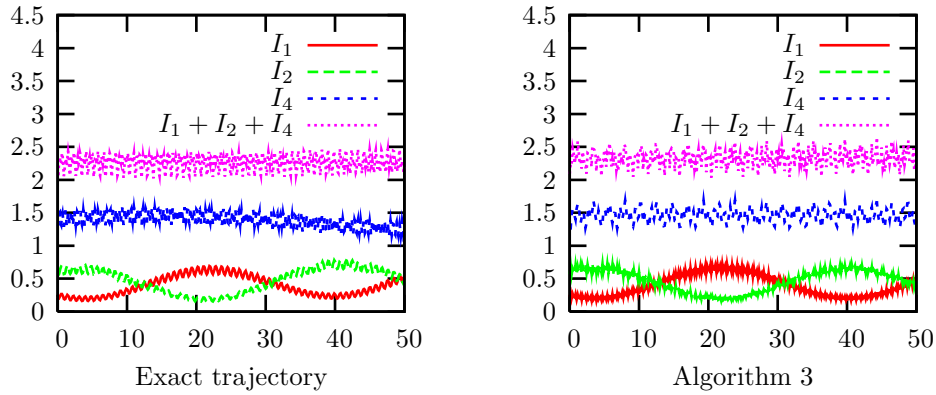

Figure 10: The exchange between I_1 and I_2 on the time scale $O(\varepsilon^{-1})$ is accurately reproduced by Algorithm 3, as well as the preservation of the adiabatic invariant $I_1 + I_2 + I_4$. The $O(\varepsilon)$ exchange between I_4 and $I_1 + I_2$ is not captured. (We take $\varepsilon = 1/70$, and we use $h = 10\varepsilon$ for Algorithm 3).

4   The extensible pendulum

In the previous sections, we have considered Hamiltonians of the form (2), where the leading order behavior of the fast variables is that of a harmonic oscillator, with either a slowly varying scalar frequency (in Section 2), or a constant matrix of fast frequencies (in Section 3). On the other hand, many examples of interest are not harmonic, and this technique cannot handle completely general fast forces. An intermediate class of system is the one discussed in [10, Sec. XIV.3], where the Hamiltonian reads

$$H(q, p) = \frac{1}{2} p^T p + V_{\text{slow}}(q) + \frac{1}{\varepsilon^2} U_{\text{fast}}(q)$$

with fast potential U_{fast} that has a change of variables $y = (y_1, y_2) = \chi(q)$ so that

$$U_{\text{fast}}(q) = \frac{1}{2} y_2^T \Omega(y_1)^2 y_2,$$

where $\Omega(y_1)$ is a $f \times f$ matrix. So, up to a change of variables, the fast potential energy is of the form considered in (2). However, the map $(q, p) \mapsto (y, p)$ is in general not symplectic. Hence we cannot

28

work with the variables (y, p), as the dynamics in these variables is not Hamiltonian. We thus need to also change variables for the momenta, so that the map $(q, p) \mapsto (y, v)$, where v are the new momenta, is symplectic, and the dynamics in (y, v) is Hamiltonian. A standard consequence of this change of variables for the momenta is that the kinetic energy turns out to depend on the positions y. The Hamiltonian reads

$$H(y, v) = T(y, v) + V_{\text{slow}}(\chi^{-1}(y)) + \frac{1}{2\varepsilon^2} y_2^T \Omega(y_1)^2 y_2. \tag{63}$$

Although the fast potential is harmonic, the fast Hamiltonian is not necessarily close to that of a harmonic oscillator.

In this section, we consider a particular case (see (64) below) that cannot be written in the form (2). After a global change of variables, in positions and momenta, we transform the Hamiltonian to the form (63) (see (65) below), where the kinetic energy turns out to still be a quadratic form of the momenta: $T(y, v) = \frac{1}{2} v^T M^{-1}(y) v$. The resulting mass matrix $M(y)$ depends on the position. We will show that our strategy can still handle this case and yields an efficient algorithm.

4.1 Transforming to internal coordinates

We consider an extensible pendulum in two dimensions (see Fig. 11). The potential energy of the system is the sum of two terms, the spring energy and a term $W(a)$ that depends on the angle a. Denote by q_x and q_y the Euclidean coordinates of the particle and by p_x and p_y the corresponding momenta. The Hamiltonian of the system reads

$$H_{\text{cartesian}}(q_x, q_y, p_x, p_y) = \frac{1}{2}(p_x^2 + p_y^2) + \frac{1}{2\varepsilon^2}(\overline{r} - r_0)^2 + W(a), \tag{64}$$

where $\overline{r} = \sqrt{q_x^2 + q_y^2}$ is the length of the spring, r_0 is the equilibrium length of the spring, ε is a small parameter, and W is a 2π-periodic non-negative function.

For illustration, this example can be considered as an oversimplified model of a molecular system, in which the spring-like potential $(2\varepsilon^2)^{-1}(\overline{r} - r_0)^2$ models each bond length between neighboring atoms, and with some potential associated with each bond angle.

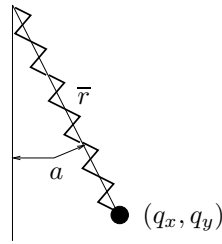
Figure 11: Extensible pendulum test-case: a particle at position (q_x, q_y) is attached to the origin using an extensible spring. We denote by $\overline{r}$ the spring length and by a the angle between the pendulum and the vertical.

The fast oscillating term $(2\varepsilon^2)^{-1}(\overline{r} - r_0)^2$ of the Hamiltonian (64) is not harmonic with respect to the *cartesian* degrees of freedom (q_x, q_y), and thus the strategy described in Section 2 cannot be directly applied to (64). However, as is often the case in molecular dynamics models, the fast oscillating term of (64) is harmonic in the *internal* degrees of freedom, the bond length $\overline{r}$ and the bond angle a. It is thus natural to introduce polar coordinates $(a, \overline{r})$, with

$$q_x = \overline{r} \cos a \quad \text{and} \quad q_y = \overline{r} \sin a,$$

so that the fast oscillating term of the potential energy is *harmonic* in these coordinates. We now introduce momenta (p_a, p_r) associated to $(a, \overline{r})$, such that the map $(q_x, q_y, p_x, p_y) \mapsto (a, \overline{r}, p_a, p_r)$ is

29

symplectic. Following [10, Example VI.5.2], this leads to choosing

$$p_r = p_x \cos a + p_y \sin a \quad \text{and} \quad p_a = -p_x \overline{r} \sin a + p_y \overline{r} \cos a.$$

In addition, rather than working with $\overline{r}$, we work in the sequel with

$$r = \overline{r} - r_0,$$

so that the fast oscillating harmonic term attains its minimum at $r = 0$, as in (2). Without loss of generality, we furthermore assume that $r_0 = 1$. The Hamiltonian then reads

$$H(a, r, p_a, p_r) = \frac{p_r^2}{2} + \frac{p_a^2}{2(1+r)^2} + \frac{1}{2\varepsilon^2} r^2 + W(a). \tag{65}$$

Although the mass matrix depends on the fast position r, we will show that our general strategy, originally developed for (2), can still be used.

Remark 4.1. As above, we choose initial conditions depending on ε such that the energy is bounded. As W is bounded from below, this implies that, at any time t, $r(t) = O(\varepsilon)$. As a consequence, one could approximate (65) by

$$H_{\text{decoup}}(a, r, p_a, p_r) = \frac{p_r^2}{2} + \frac{p_a^2}{2} + \frac{1}{2\varepsilon^2} r^2 + W(a).$$

In this case, (r, p_r) and (a, p_a) are decoupled. In the following, we do not make this approximation, and we work with (65).

4.2 Derivation of the symplectic scheme

Starting from (65), we now follow our usual strategy: we first precondition the fast variables, and next consider the Hamilton-Jacobi form of the equations. Observe that, in the case at hand, the fast frequency is constant. We thus follow the approach described in [12] and consider the time-dependent change of variable $(r, p_r) \mapsto (\overline{b}, p_b)$ defined by

$$r = \overline{b} \cos \frac{t}{\varepsilon} + \varepsilon p_b \sin \frac{t}{\varepsilon} \quad \text{and} \quad p_r = -\frac{\overline{b}}{\varepsilon} \sin \frac{t}{\varepsilon} + p_b \cos \frac{t}{\varepsilon}. \tag{66}$$

In the variables $(a, \overline{b}, p_a, p_b)$, the dynamics is again a Hamiltonian dynamics for the Hamiltonian $H_{\text{pend}} \left(\frac{t}{\varepsilon}, a, b, p_a, p_b \right)$, with

$$H_{\text{pend}}(\tau, a, \overline{b}, p_a, p_b) = \frac{p_a^2}{2(1 + \overline{b} \cos \tau + \varepsilon p_b \sin \tau)^2} + W(a).$$

Let $\overline{S}_\varepsilon(t, a, \overline{b}, P_a, P_b)$ solve the Hamilton-Jacobi equation associated to H_{pend}. As r and $\overline{b}$ are of order ε, we make the change of variables and of unknown function:

$$b = \frac{\overline{b}}{\varepsilon} \quad \text{and} \quad S_\varepsilon(t, a, b, P_a, P_b) = \overline{S}_\varepsilon(t, a, \varepsilon b, P_a, P_b),$$

so that S_ε satisfies

$$\partial_t S_\varepsilon = H_{\text{pend}} \left(\frac{t}{\varepsilon}, a + \partial_{P_a} S_\varepsilon, \varepsilon b + \partial_{P_b} S_\varepsilon, P_a, P_b \right), \quad S_\varepsilon(0, a, b, P_a, P_b) = 0. \tag{67}$$

We make the ansatz

$$\begin{aligned} S_\varepsilon(t, a, b, P_a, P_b) &= S_0(t, \tau, a, b, P_a, P_b) + \varepsilon S_1(t, \tau, a, b, P_a, P_b) \\ &+ \text{ higher order terms in } \varepsilon^k, \ k \geq 2, \end{aligned} \tag{68}$$

30

where the fast time τ is defined by

$$\tau = \frac{t}{\varepsilon},$$

and where the functions $(S_k)_{k \geq 0}$ are assumed to be 2π periodic in τ. We now insert (68) in (67), identify the first variable of H_{pend} with the fast time τ, and expand in powers of ε.

The identification follows the same lines as in the previous sections. We obtain that S_0 is independent of b, P_b and τ, and satisfies $S_0(t) = S_0^{\mathrm{SE}}(t) + O(t^2)$ with

$$S_0^{\mathrm{SE}}(t, a, P_a) = t \left(\frac{P_a^2}{2} + W(a) \right), \tag{69}$$

while

$$S_1 \equiv 0. \tag{70}$$

Using the method of characteristics, we obtain that $S_2(t) = S_2^{\mathrm{SE}}(t) + O(t^2)$, with

$$
\begin{aligned}
S_2^{\mathrm{SE}}(t, \tau, a, b, P_a, P_b) &= P_a^2 \left[P_b \cos \tau - b \sin \tau \right] - P_b (P_a + t W'(a))^2 \\
&+ t \left[\frac{3}{4}(b^2 + P_b^2)(P_a + t W'(a))^2 - \frac{1}{2}(P_a + t W'(a))^4 \right].
\end{aligned}
\tag{71}
$$

Collecting (69), (70) and (71), we obtain the following approximation of the generating function:

$$S_\varepsilon(h, a, b, P_a, P_b) \approx \widetilde{S_\varepsilon}(h, a, b, P_a, P_b) = S_0^{\mathrm{SE}}(h, a, P_a) + \varepsilon^2 S_2^{\mathrm{SE}}(h, \frac{h}{\varepsilon}, a, b, P_a, P_b).$$

Returning to the variables (a, r, p_a, p_r) of the Hamiltonian in (65), we obtain a symplectic scheme, called Algorithm 4 in the sequel, which we denote by $(a^{n+1}, r^{n+1}, p_a^{n+1}, p_r^{n+1}) = \Psi_h^{\mathrm{pendulum}}(a^n, r^n, p_a^n, p_r^n)$.

Algorithm 4 (Symplectic scheme $\Psi_h^{\mathrm{pendulum}}(a, r, p_a, p_r)$). Set $(a, r, p_a, p_r) = (a^n, r^n, p_a^n, p_r^n)$, $\tau = h/\varepsilon$ and perform the following steps:

1. Change of variables: set $\quad b = r/\varepsilon, \quad p_b = p_r$.

2. Solve for (P_a, P_b) in the equations

$$
\begin{cases}
p_b &= P_b - \varepsilon P_a^2 \sin \tau + \dfrac{3}{2} h \varepsilon b \left(P_a + h W'(a) \right)^2, \\[2mm]
p_a &= P_a + h W'(a) - 2 h \varepsilon^2 P_b \left(P_a + h W'(a) \right) W''(a) \\[2mm]
&+ h^2 \varepsilon^2 \left[\dfrac{3}{2}(b^2 + P_b^2)(P_a + h W'(a)) - 2(P_a + h W'(a))^3 \right] W''(a).
\end{cases}
$$

3. Set

$$
\begin{aligned}
A &= a + h P_a + 2\varepsilon^2 P_a (P_b \cos \tau - b \sin \tau) - 2\varepsilon^2 P_b (P_a + h W'(a)) \\[2mm]
&+ h \varepsilon^2 \left[\frac{3}{2}(b^2 + P_b^2)(P_a + h W'(a)) - 2(P_a + h W'(a))^3 \right].
\end{aligned}
$$

4. Set $B = b + \varepsilon P_a^2 \cos \tau - \varepsilon (P_a + h W'(a))^2 + \dfrac{3}{2} h \varepsilon P_b (P_a + h W'(a))^2$.

5. Return to the original variables: $\quad R = \varepsilon B \cos \tau + \varepsilon P_b \sin \tau, \quad P_r = -B \sin \tau + P_b \cos \tau$.

Set $\left(a^{n+1}, r^{n+1}, p_a^{n+1}, p_r^{n+1} \right) = (A, R, P_a, P_r)$.

To solve the implicit equations for (P_a, P_b), we first observe that, once P_a is known, P_b can be explicitly determined. We thus first recast the problem as a scalar implicit equation on P_a and solve it using a fixed point algorithm before determining P_b.

Neglecting all terms of order ε^3, the scheme Ψ_h^{pendulum} is first order in h. As in Section 3.1, we use Ψ_h^{pendulum} to build a symplectic and symmetric scheme, denoted Algorithm 5 in the sequel, following

$$(A, R, P_a, P_r) = \Psi_h^{\text{pend.sym.}}(a, r, p_a, p_r) = \left(\Psi_{h/2}^{\text{pendulum}}\right)^* \Psi_{h/2}^{\text{pendulum}}(a, r, p_a, p_r),$$

where $\left(\Psi_h^{\text{pendulum}}\right)^*$ denotes the the adjoint method, which is denoted Algorithm 6.

Algorithm 5 (Symmetric Symplectic Scheme $\Psi_h^{\text{pend.sym.}}(a, r, p_a, p_r)$). Set $(a, r, p_a, p_r) = (a^n, r^n, p_a^n, p_r^n)$ and perform the following steps:

1. Set $(\overline{A}, \overline{R}, \overline{P}_a, \overline{P}_r) = \Psi_{h/2}^{\text{pendulum}}(a, r, p_a, p_r)$.

2. Set $(A, R, P_a, P_r) = \left(\Psi_{h/2}^{\text{pendulum}}\right)^* (\overline{A}, \overline{R}, \overline{P}_a, \overline{P}_r)$.

Set $\left(a^{n+1}, r^{n+1}, p_a^{n+1}, p_r^{n+1}\right) = (A, R, P_a, P_r)$.

Algorithm 6 (Symplectic Scheme $\left(\Psi_h^{\text{pendulum}}\right)^* (a, r, p_a, p_r)$). Set $(a, r, p_a, p_r) = (a^n, r^n, p_a^n, p_r^n)$, $\tau = h/\varepsilon$ and perform the following steps:

1. Rotate the variables: set $b = \dfrac{r}{\varepsilon} \cos \tau + p_r \sin \tau, \quad p_b = -\dfrac{r}{\varepsilon} \sin \tau + p_r \cos \tau$.

2. Solve for (A, B) in the equations

$$\left\{ \begin{aligned} a \quad = \quad & A - hp_a + 2\varepsilon^2 p_a(p_b \cos \tau + B \sin \tau) - 2\varepsilon^2 p_b(p_a - hW'(A)) \\ & - h\varepsilon^2 \left[\frac{3}{2}(B^2 + p_b^2)(p_a - hW'(A)) - 2(p_a - hW'(A))^3\right], \\ b \quad = \quad & B + \varepsilon p_a^2 \cos \tau - \varepsilon(p_a - hW'(A))^2 - \frac{3}{2}h\varepsilon p_b(p_a - hW'(A))^2. \end{aligned} \right.$$

3. Set $P_b = p_b + \varepsilon p_a^2 \sin \tau - \dfrac{3}{2}h\varepsilon B(p_a - hW'(A))^2$.

4. Set

$$\begin{aligned} P_a \quad = \quad & p_a - hW'(A) + 2h\varepsilon^2 p_b(p_a - hW'(A))W''(A) \\ & + h^2\varepsilon^2 \left[\frac{3}{2}(B^2 + p_b^2)(p_a - hW'(A)) - 2(p_a - hW'(A))^3\right] W''(A). \end{aligned}$$

5. Return to the original variables: $R = \varepsilon B, \quad P_r = P_b$.

Set $\left(a^{n+1}, r^{n+1}, p_a^{n+1}, p_r^{n+1}\right) = (A, R, P_a, P_r)$.

4.3   Numerical results

We have implemented Algorithm 5 on the Hamiltonian (65) with

$$W(a) = (\cos a)^2.$$

We first choose $\varepsilon = 2.10^{-3}$ and $h = 0.02$, and monitor the evolution of the energy and of the adiabatic invariant

$$I = \frac{p_r^2}{2} + \frac{r^2}{2\varepsilon^2} \tag{72}$$

up to time $T = 10^6$. Results are shown on Fig. 12. We observe no drift.

We next study the robustness of the algorithm as ε decreases. We set the time step to $h = 0.02$, and consider the variations (53) of the energy and of the adiabatic invariant (72), over the time interval $t \in [0, 10^4]$, for stiffness ε varying between 10^{-3} to 1. Results are shown in Fig. 12. Again, the algorithm performs equally well when ε decreases to 0, up to extremely peaked resonances in the energy preservation.
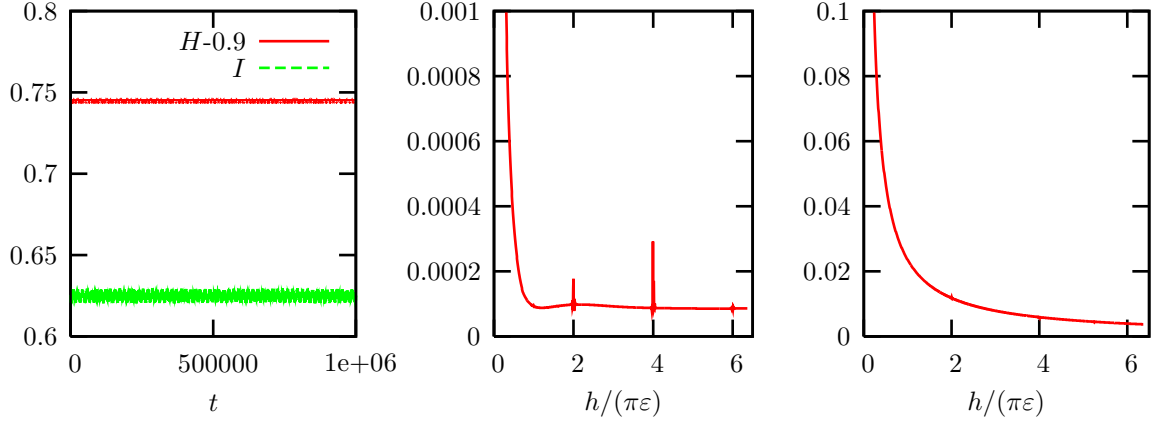

Figure 12: Left: energy (for convenience, we plot $H - 0.9$) and adiabatic invariant I along the trajectory computed with Algorithm 5 ($\varepsilon = 2.10^{-3}$ and $h = 0.02$). Center and right: maximum variations (53) of the energy (center) and of the adiabatic invariant I (right) on the time interval $[0, 10^4]$, for several ε ($h = 0.02$), for Algorithm 5.

Acknowledgments

References

[1] F. Bornemann. *Homogenization in time of singularly perturbed mechanical systems.* Number 1687 in Lecture Notes in Mathematics. Springer, Berlin, 1998.

[2] F. A. Bornemann and C. Schütte. Homogenization of Hamiltonian systems with a strong constraining potential. *Phys. D*, 102(1-2):57–77, 1997.

[3] F. Castella, P. Chartier, and E. Faou. An averaging technique for highly oscillatory Hamiltonian problems. *SIAM J. Numer. Anal.*, 47(4):2808–2837, 2009.

[4] D. Cohen, T. Jahnke, K. Lorenz, and C. Lubich. Numerical integrators for highly oscillatory Hamiltonian systems: a review. In *Analysis, modeling and simulation of multiscale problems*, pages 553–576. Springer, Berlin, 2006.

[5] M. Dobson, C. Le Bris, and F. Legoll. Symplectic schemes for highly oscillatory Hamiltonian systems with varying fast frequencies. *C. R. Acad. Sci. Paris*, submitted.

[6] K. Feng. Difference schemes for Hamiltonian formalism and symplectic geometry. *J. Comput. Math.*, 4(3):279–289, 1986.

[7] B. García-Archilla, J. M. Sanz-Serna, and R. D. Skeel. Long-time-step methods for oscillatory differential equations. *SIAM J. Sci. Comput.*, 20(3):930–963 (electronic), 1999.

[8] V. Grimm and M. Hochbruck. Error analysis of exponential integrators for oscillatory second-order differential equations. *J. Phys. A*, 39(19):5495–5507, 2006.

[9] H. Grubmüller, H. Heller, A. Windemuth, and K. Schulten. Generalized Verlet algorithm for efficient molecular dynamics simulations with long range interaction. *Mol. Sim.*, 6(1-3):121–142, 1991.

[10] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration*, volume 31 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, second edition, 2006.

[11] C. Le Bris and F. Legoll. Derivation of symplectic numerical schemes for highly oscillatory Hamiltonian systems. *C. R. Acad. Sci. Paris*, 344(4):277–282, 2007.

[12] C. Le Bris and F. Legoll. Integrators for highly oscillatory Hamiltonian systems: an homogenization approach. *Discrete Cont. Dyn-B*, 13(2):347–373, 2010.

[13] J. M. Sanz-Serna. Mollified impulse methods for highly oscillatory differential equations. *SIAM J. Numer. Anal.*, 46(2):1040–1059, 2008.

[14] M. Tao, H. Owhadi, and J. E. Marsden. Nonintrusive and structure preserving multiscale integration of stiff odes, sdes, and hamiltonian systems with hidden slow dynamics via flow averaging. *Multiscale Modeling & Simulation*, 8(4):1269–1324, 2010.

[15] M. Tuckerman, B. Berne, and G. Martyna. Reversible multiple time scale molecular dynamics. *J. Chem. Phys.*, 97(3):1990–2001, 1992.